Universidade de Aveiro
Departamento de Electrónica, Telecomunicações e Informática,
2017

Maksym Volodymyrovych Ketsmur

Sistema de resposta automática a questões em português

Maksym Volodymyrovych Ketsmur

Sistema de resposta automática a questões em português

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Informática, realizada sob a orientação científica do Doutor António Teixeira, Professor Associado do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro e do Doutor Mário Rodrigues Professor Adjunto da Escola Superior de Tecnologia e Gestão de Águeda da Universidade de Aveiro.

o júri / the juri

presidente / president

Joaquim Arnaldo Carvalho Martins
Professor Catedrático, Universidade de Aveiro

vogais / examiners committee

Alberto Manuel Brandão Simões
Professor Adjunto, Instituto Politécnico do Cávado e do Ave

António Joaquim da Silva Teixeira (Orientador)
Professor Associado, Universidade de Aveiro

agradecimentos

Expresso e registo o meu profundo reconhecimento e gratidão:

Ao orientador Prof. António Joaquim da Silva Teixeira e ao co-orientador Prof. Mário Jorge Ferreira Rodrigues, pelas suas colaborações interessadas e intensivas. As brilhantes e sábias sugestões e a sua total disponibilidade que demonstraram ao longo do desenvolvimento deste trabalho.

Aos meus colegas e amigos do Departamento de Eletrónica, Telecomunicações e Informática e do Instituto de Engenharia Electrónica e Telemática de Aveiro pelo apoio e incentivo.

A todos os meus familiares que diariamente contribuíram com toda a paciência e a compreensão, em especial, aos meus pais Volodymyr e Olena e as minhas irmãs Valeriya e Viktoriya.

Finalmente, um agradecimento muito especial à minha namorada Joana, pelo todo apoio, compreensão, carinho e amor.

A todos aqui deixo a minha mais profunda gratidão.

Palavras-Chave

Sistemas de resposta a questões, Processamento de Linguagem Natural, Web Semântica, Questões factuais, Entidades Mencionadas.

Resumo

Atualmente, a forma predominante de pesquisas efetuadas na Internet por utilizadores comuns consiste na utilização de palavras-chave. Contudo, a utilização desta forma de pesquisa de informação é muitas vezes inadequada para expressar a verdadeira intenção do utilizador. Desta forma, surge a necessidade de desenvolvimento de sistemas de resposta automática a questões.

O objetivo deste trabalho consiste no desenvolvimento de um sistema de resposta a questões efetuadas em linguagem natural portuguesa.

O trabalho realizado consistiu em duas abordagens distintas, sendo que a segunda representa a versão aperfeiçoada da primeira.

O sistema inclui diferentes módulos que permitem, em primeiro lugar, efetuar o processamento de linguagem natural de entrada com objetivo de identificar a intenção do utilizador, de seguida uma pesquisa semântica é efetuada com objetivo de obter a informação necessária para a resposta a pergunta e, por fim, a resposta é gerada e devolvida ao utilizador.

Os resultados obtidos demonstraram que o sistema é capaz de identificar a verdadeira intenção do utilizador e apresentar resposta à maioria das questões factuais que possuem informação necessária para a resposta na fonte de dados semântica.

As duas variantes do sistema apresentaram bons resultados na resposta a questões factuais, sendo que existem algumas limitações tanto na identificação de entidades na questão de entrada, como na ausência de informação necessária para a resposta.

Keywords Question Answering, Semantic Web, Natural Language Processing, Factual questions, Named Entities.

Abstract The predominant form of searches performed on the Internet is the use of keywords. However, the use of this form of information search is often inadvertent to express the true intent of the user. In this way, the need to develop automatic question answering systems arises.
The aim of this work is to develop a question answering system to respond to questions made in Portuguese natural language.
The work carried out consisted in two distinct approaches in which the second represents the improved version of the first.
The system includes different modules that allow, in the fisrt place, perform the natural language processing of input question in order to identify the user's intention, then a semantic search is performed in order to obtain the needed information to answer the question and finally response is generated and returned to the user.
The obtained results show that the system is able to identify the true intention of the user and to answer most of the factual questions that have the answer in the semantic data source.
The two approaches of the system presented good capability in answering factual questions, being that there are some limitations both in the identification of entities in the entry question, and the lack of information needed for the response.

Conteúdo

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introdução

1.1 Interfaces de Linguagem Natural

Atualmente, a forma predominante de pesquisas efetuadas na Internet e que já se tornou familiar para os utilizadores comuns consiste na utilização de palavras-chave. Contudo, a utilização desta forma de pesquisa de informação é muitas vezes inadequada para expressar a verdadeira intenção do utilizador (Song et al. (2015)).

Pesquisas efetuadas utilizando palavras-chave retornam uma extensa lista de resultados possíveis, na qual o utilizador é obrigado a procurar a informação pretendida. Por um lado, a maior quantidade de informação permite comparar os resultados, mas por outro torna as pesquisas mais difíceis e, tendo em conta a quantidade e o atual crescimento de informação online, a necessidade de sistemas que permitam ao utilizador comum efetuar pesquisas estruturadas torna-se cada vez mais necessitada (Hirschman and Gaizauskas (2001); Navigli and Ponzetto (2012)).

Com a utilização de pesquisas bem estruturadas, torna-se possível entender melhor a intenção do utilizador, devolvendo-lhe o resultado pretendido mais correto possível em vez de uma lista. Com isto, surgiu a necessidade de desenvolver novos sistemas de pesquisa semântica, possibilitando ao utilizador efetuar uma pesquisa estruturada utilizando a linguagem natural, que permite identificar a intenção do utilizador, e a semântica, que permite efetuar a pesquisa em diferentes fontes.

As interfaces de linguagem natural são um tipo de interface humano-computador que aceita entradas em linguagem natural e identifica a intenção do utilizador, com recurso à análise de elementos como verbos e nomes, para decidir ações como criar, selecionar ou modificar dados em aplicações de software.

Resposta automática a questões é a área ativa de estudo na área de processamento de linguagem natural e linguística computacional, sendo um dos seus principais desafios a compreensão generalizada de entradas que podem ser ambíguas. Uma interface de linguagem natural geral intuitiva é um dos objetivos ativos da Web Semântica.

1.2 Objetivos

O principal objetivo deste trabalho é o desenvolvimento de um sistema de resposta a questões efetuadas em linguagem natural.

O sistema deverá ser capaz de receber as questões do utilizador em linguagem natural portuguesa, processá-las e devolver uma resposta válida às mesmas.

Para alcançar os objetivos, o sistema tem de efetuar o processamento da questão com o objetivo de entender a intenção do utilizador, e, para além deste aspeto, o sistema tem de implementar módulos que permitam efetuar uma classificação do tipo de pergunta e de resposta, permitindo assim obter melhores resultados finais.

O sistema a desenvolver terá que utilizar técnicas de identificação de entidades mencionadas e conceitos na questão do utilizador, efetuando mais um passo na identificação da intenção do utilizador.

Terá de ser desenvolvido o módulo de geração da consulta à base de dados tendo em conta a pergunta do utilizador, ou seja, desenvolver técnicas automáticas que permitam traduzir a questão do utilizador para linguagem semântica.

Por fim, o sistema terá de implementar o módulo de geração de resposta que agregará toda a informação resultante dos módulos anteriores e devolver uma resposta ao utilizador.

1.3 Estrutura da dissertação

O documento está dividido em 6 capítulos que têm como objetivo explicar os objetivos do trabalho, dar uma breve introdução relativamente a sistemas de resposta automática a questões, apresentar os estudos já efetuados nesta área, sistemas já existentes, discutir todas as fases de desenvolvimento do sistema em questão e, por fim, apresentar os resultados obtidos.

No segundo capítulo intitulado *Trabalho relacionado e Estado de arte* são abordados conceitos base relacionados com resposta automática a questões e o processamento de linguagem natural. Apresenta-se um conjunto de sistemas já existentes na área, as suas vantagens e desvantagens e, por fim, uma conclusão geral sobre a pesquisa efetuada.

O terceiro capítulo intitulado *Recursos e Ferramentas utilizados* falará sobre as diferentes ferramentas que permitem estabelecer uma forma de entendimento entre o utilizador e o sistema com objetivo de analisar a questão do utilizador e entender a sua intenção. As diferentes fontes de informação semântica também são discutidas neste capitulo, visto serem o recurso principal para dar resposta a questões.

No quarto capítulo é apresentada a arquitetura geral de sistemas de resposta a questões, assim como os módulos essenciais ao funcionamento básico de sistema e o objetivo de cada um. No quinto capítulo são apresentados de forma pormenorizada os passos de desenvolvimento do sistema, nomeadamente as duas variantes desenvolvidas, a sua avaliação, limitações, resultados obtidos e por fim uma comparação entre as mesmas. O capítulo final demonstra os principais resultados obtidos com a segunda variante do sistema, possíveis melhorias e uma conclusão final sobre o trabalho efetuado.

Capítulo 2

Trabalho Relacionado e Estado da Arte

2.1 Introdução

Neste capítulo são abordados os conceitos base relacionados com o tema de resposta automática a questões, apresentam-se alguns sistemas relacionados, a sua descrição e as principais críticas.

Começamos por conceitos base de Processamento de Linguagem Natural (PLN), visto ser o processamento essencial na compreensão do utilizador, ou seja, identificação da sua intenção.

Após são abordados os conceitos da área de Web Semântica (ontologia, *synsets*, etc.), importantes na identificação de entidades e conceitos no mundo semântico.

Por fim, são identificados e analisadas diferentes técnicas de classificação do tipo de pergunta e de resposta, apresentando as suas características.

2.2 Conceitos base

2.2.1 Processamento de Linguagem Natural

Processamento de Linguagem Natural (PLN) é a tarefa fundamental na compreensão do texto de entrada, identificando a classe gramatical das palavras, relações existentes entre estas, entidades e conceitos que possam estar presentes, permitindo compreender a questão e identificar a intenção do utilizador. Desta forma, a capacidade de identificação da intenção do utilizador consiste na análise da questão de forma semelhante ao humano, utilizando diferentes técnicas que permitem identificar entidades mencionadas, conceitos, ações, ligações entre as entidades, etc. (Hirschman and Gaizauskas (2001); Liddy (2001)).

Um vasto número de avaliações de sistemas de pesquisa semântica referem a preferência do utilizador na utilização de linguagem natural livre durante a pesquisa em vez de entradas controladas ou pesquisas baseadas em visualização. Embora a flexibilidade oferecida por esta abordagem seja uma vantagem significativa, também se poderá tornar numa grande complexidade (Kumar et al. (2016)). Permitir aos utilizadores uma liberdade total na

escolha de termos aumenta a dificuldade destas ferramentas a desambiguar a pesquisa e entender a sua verdadeira intenção (Elbedweihy et al. (2013)).

A desambiguação do sentido da palavra tem como objetivo entender o que os termos significam. Enquanto que para os humanos é fácil entender o sentido da palavra, para a maquina é uma tarefa difícil (Elbedweihy et al. (2013); Guha et al. (2003)). Desta forma, para a utilização da pesquisa semântica é essencial a utilização de ferramentas de desambiguação de termos, visto que a pesquisa semântica descreve o processo de correspondência entre o conteúdo do documento e a intenção do utilizador (Hoffart et al. (2011)).

Os sistemas de pesquisa semântica adotam diferentes abordagens de pesquisa, que vão desde a linguagem natural (livre ou guiada) até às interfaces baseadas em visualização (formulários ou grafos). Cada uma destas estratégias fornece diferentes níveis de flexibilidade ao utilizador, expressão de linguagem de consulta e suporte durante a formulação da consulta (Elbedweihy et al. (2013)). Contudo, como mencionado anteriormente, a flexibilidade dificulta o mapeamento dos termos com os conceitos, propriedades e entidades ontológicas.

Uma destas dificuldades é a polissemia (uma palavra com mais do que um significado) e a sinonímia (múltiplas palavras com o mesmo significado). Enquanto a primeira afeta a precisão, fornecendo correspondências falsas, a segunda afeta o *recall* por causar a falta de correspondências semânticas verdadeiras. Desta forma, ambas recorrem às técnicas de desambiguação do sentido da palavra (Elbedweihy et al. (2013)).

O PLN consiste num conjunto de sub-processos que variam consoante a necessidade de cada sistema, mas existe um *pipeline* clássico que consiste nas seguintes sub-tarefas:

- Tokenização - parte o texto em frases e palavras e efetua a lematização.
- Análise sintática - marcação parcial, *stemming* e *NER*.
- Sintaxe - análise de dependências
- Semântica - resolução de correferência e desambiguação do sentido da palavra.

2.2.1.1 Análise sintática

A parte essencial da análise sintática é a marcação parcial (*POS*), sendo uma forma básica desta análise, que possui inúmeras aplicações PLN (Gimpel et al. (2011)).

Classes gramaticais (*Parts of speech*) foram reconhecidas em linguística há muito tempo e podem ser distinguidas entre vários tipos (Tabela 2.1).

A marcação parcial consiste na atribuição automática de classes gramaticais para os *tokens* (termos) de entrada, resultantes da tarefa de tokenização (Voutilainen (2003)).

2.2.1.2 Entidades Mencionadas

Entidades mencionadas são frases que contêm nomes de pessoas, organizações ou locais. A tarefa de identificação de entidades mencionadas é uma das tarefas importantes da área de extração de informação. Atualmente existe muita contribuição no reconhecimento de

Tag	Class	Tag	Class	Tag	Class
ADJ	Adjetivo	ADP	Preposição	ADV	Advérbio
AUX	Auxiliar	CCONJ	Conjunção	DET	Determinador
INTJ	Interjeição	NOUN	Substantivo	NUM	Numeral
PART		PRON	Pronome	PROPN	Nome próprio
PUNCT	Pontuações	SCONJ		SYM	Símbolo
VERB	Verbo	X	outros		

Tabela 2.1: Principais classes gramaticais (Fonte: http://universaldependencies.org/u/pos/index.html)

entidades mencionadas, especialmente para o inglês (Tjong Kim Sang and De Meulder (2003)).

2.2.2 Web Semântica

Web Semântica é uma forma de conteúdo da Web que é compreendida por pessoas e computadores da mesma forma, criando uma estrutura do conteúdo significativo das páginas Web. A web Semântica não é uma Web à parte, mas sim uma extensão da atual, na qual a informação tem um significado bem definido, permitindo às pessoas e computadores trabalharem em cooperação (Berners-Lee et al. (2001)).

Cada vez mais a informação é disponibilizada em forma de *Linked Data*, que consiste no uso da Web para interligar diferentes fontes de informação. Estas podem ser tão diversas quanto as bases de dados de duas diferentes organizações em diferentes localizações geográficas. Tecnicamente, Linked Data refere-se aos dados publicados na Internet de forma a que possam ser compreendidos tanto pelos humanos como por computadores (Damljanovic et al. (2011); Unger et al. (2012)).

Os dados na web semântica são frequentemente publicados em forma *RDF* (*Resource Description Framework*), que fornece um padrão de expressar grafos de dados e compartilhá-los com outras pessoas e, talvez mais importante, com máquinas. Por ser uma "recomendação"do W3C, uma grande coleção de ferramentas e serviços surgiu em volta do *RDF* (Bizer et al. (2009); Segaran et al. (2009)).

RDF é uma linguagem para expressar modelos de dados usando triplos, que são constituídos por sujeito, predicado e objeto. Além disso, esta linguagem acrescenta vários conceitos importantes que tornam esses modelos muito mais precisos, robustos e, o mais importante, removem a ambiguidade ao transmitir dados semânticos entre máquinas (Segaran et al. (2009); Berners-Lee et al. (2001)).

Para evitar este tipo de ambiguidades, *RDF* trata tudo como um recurso. Um recurso é simplesmente algo que pode ser identificado como *URI* (*Universal Resourse Identifier*). Um dos exemplos mais familiares são os *URL* (*Universal Resourse Locators*) usados para identificar de forma única as páginas web. *URLs* são um subconjunto de *URIs* que identificam onde é que a informação digital pode ser recuperada. Enquanto *URLs* dizem onde a informação pode ser encontrada, *URI*s generalizam o conceito de *URLs*, dizendo que algo,

recuperado eletronicamente ou não, pode ser identificado de uma forma única.

2.2.2.1 *Synsets*

Synset é um conjunto não ordenado de palavras e frases cognitivamente sinónimas. Cada membro de um determinado *synset* expressa o mesmo conceito, permitindo que todos os membros do *synset* sejam intercambiáveis em todos os contextos (por exemplo Carro e Automóvel) (Fellbaum (1998)).

Relativamente a itens lexicais, estes podem ter várias relações entre si:

Sinónimos - são palavras que têm o mesmo significado (por exemplo observar, examinar, considerar)

Polissemia - Palavras polissémicas são aquelas que representam diferentes significados (por exemplo manga, que pode significar parte da roupa ou fruta)

Hipónimos - Por exemplo, rosa é um hipónimo de flor em que as rosas são tipos de flores. Por outras palavras, se X é um hipónimo de Y, então a extensão de X é um subconjunto da extensão de Y. Assim, podemos dizer que a hiponimia é uma relação de inclusão (Murphy (2006); Fellbaum (1998)).

Hiperónimos - Hiperónimos é a relação inversa da hiponimia. Ou seja, tendo em conta o exemplo anterior, onde rosa era hipónimo da flor, neste caso flor é um hiperónimo de rosa (Murphy (2006)).

2.2.2.2 Ontologia

Na filosofia, uma ontologia é uma teoria sobre a natureza de tipo de coisas existentes, enquanto que na Web Semântica uma ontologia é um documento ou arquivo que define formalmente as relações entre os termos.

Tipicamente a ontologia para a Web tem uma taxonomia e um conjunto de regras de inferência. A taxonomia define classes de objetos e relações entre eles (por exemplo, um endereço pode ser definido como um tipo de localização). Com isto é possível expressar um grande número de relações entre as entidades, atribuindo propriedades às classes e permitindo que as subclasses herdem essas propriedades.

As regras de inferência nas ontologias fornecem poder adicionar. Uma ontologia pode expressar a regra "Se um código de cidade é associado à um código de estado e um endereço usa esse código da cidade, esse endereço possui o código de estado associado."Berners-Lee et al. (2001).

Desde que *URIs* identifiquem tudo como um recurso, o sujeito e o objeto de uma declaração *RDF* podem ser um recurso e os predicados são sempre recursos.

2.2.2.3 Triplos

Como referido anteriormente, o significado das paginas Web é expresso pelo *RDF*, que o codifica num conjunto de triplos, sendo que cada triplo contém um sujeito, um predicado/verbo e um objeto. Estes triplos podem ser representados usando *tags XML*. No *RDF*, um documento faz afirmações de que coisas particulares (pessoas, páginas Web ou algo) possuem propriedades (como "é uma irmã de", "nasceu em") com certos valores (outra pessoa, local, outra página Web).

Esta estrutura é uma forma natural de descrever a grande maioria dos dados processados pelas máquinas. Sujeito e objeto são identificados por um identificador universal de recursos (*URI*). Os predicados/verbos também são identificados por *URIs*, o que permite que qualquer pessoa defina um novo conceito, um novo predicado, apenas definindo um *URI* para o mesmo.

2.3 Perguntas e respostas

O primeiro passo do desenvolvimento de sistemas de resposta a questões consiste em perceber como é que os utilizadores efetuam as pesquisas e que tipo de respostas esperam (Hirschman and Gaizauskas (2001)).

2.3.1 Tipos de perguntas

Durante a pesquisa efetuada foram identificados diferentes tipos de perguntas que podem ser feitas na utilização de sistemas de pesquisa semântica, sendo que as mesmas são distinguidas consoante o tipo de resposta – factual, opinião, resumo, etc... (Hirschman and Gaizauskas (2001)).

2.3.1.1 Perguntas factuais (O quê, Quando, Qual, Quem, Como)

Estas perguntas são baseadas em factos que exigem uma frase curta como resposta, por exemplo, *Quem é o realizador do filme XPTO?*. Este tipo de perguntas geralmente começam com um advérbio "Qu*", têm como resposta uma entidade, que pode ser facilmente obtida a partir de bases de dados semânticas mais conhecidas, como *DBpedia*, *Wikipedia*, *WikiData*, etc..

Os sistemas de resposta atuais têm um bom desempenho ao responder a perguntas deste tipo, uma vez que não exigem um processamento complexo de linguagem natural para conseguir a resposta. Contudo apesar de serem de tratamento fácil, a maior dificuldade consiste na identificação do tipo de pergunta factual, sendo que estas se dividem nos seguintes tipos:

Descritivas - Este tipo de perguntas exige encontrar a definição ou descrição do termo (evento ou entidade) na questão. Normalmente este tipo de perguntas começa com "*O que é*". É difícil encontrar um tipo de resposta para estas perguntas, visto que se podem referir a qualquer evento ou entidade.

Difusas - Perguntas que não trazem informação suficiente para conseguir responder à mesma, por exemplo "*Liste todas as pessoas altas da cidade.*".

Relação - Consiste na identificação da relação entre as diferentes entidades mencionadas (por exemplo: *Bear Grylls trabalha no Discovery?* onde *Bear Grylls* é uma entidade mencionada do tipo *empregado* e *Discovery* é uma entidade mencionada do tipo *empresa*). Estas perguntas requerem a identificação das entidades mencionadas, a resolução de correferência, a extração de relação, etc..

Diálogo - São geralmente perguntas incompletas e sintaticamente incorretas que tornam difícil identificar a intenção do utilizador na pergunta.

Mal formuladas ou ambíguas - Perguntas com erros ortográficos ou ambíguas, para as quais dificilmente se conseguem obter respostas.

2.3.1.2 Perguntas de listagem

Perguntas de listagem requerem a lista de entidades ou factos na resposta, como por exemplo "*Liste o nome dos empregados que ganham mais de 5 mil.*". Os sistemas consideram este tipo de perguntas como uma série de questões factuais, que são feitas sequencialmente, ignorando as respostas anteriores. Tendo em conta que este tipo de perguntas é factual, um dos problemas consiste em definir um número razoável de entidades a devolver para o utilizador.

2.3.1.3 Perguntas hipotéticas

Procuram informações relacionadas com qualquer evento hipotético. Perguntas deste tipo geralmente começam com "O que aconteceria se". Exigem técnicas de recuperação de informação para gerar respostas, sendo que estas são subjetivas.

2.3.1.4 Perguntas causais

Perguntas causais (como, porquê) requerem uma explicação sobre uma entidade. Ao contrário das perguntas factuais, onde as características da entidade mencionada é a resposta, este tipo de perguntas exige uma análise do texto na qual a entidade mencionada está referida para gerar a resposta. Um dos maiores problemas consiste na identificação da resposta mais significante, bem como identificar a verdadeira intenção da pergunta, por exemplo "*Porquê ele comprou aquele carro?*"pode ter várias interpretações como: "*Porquê comprou?*", "*Porquê carro?*", "*Porquê ele?*".

2.3.1.5 Perguntas de confirmação

Este tipo de perguntas requerem respostas do tipo "Sim" ou "Não". São necessários mecanismos de inferência, conhecimento mundial e raciocínio de senso comum para conseguir gerar respostas.

2.3.2 Tipos de respostas

Tal como as perguntas, as respostas diferem, podendo ser curtas, longas, listas ou narrativas. Por exemplo, se um utilizador pretende uma justificação, isso requer uma resposta longa (Hirschman and Gaizauskas (2001)).

As respostas podem ser formadas de duas formas – extração ou geração. Na primeira os fragmentos dos documentos originais que contêm a resposta são recortados e apresentados ao utilizador, enquanto que na segunda a resposta é extraída de múltiplas frases ou de vários documentos (Hirschman and Gaizauskas (2001)).

2.4 Resposta automática a questões

Question Answering é uma área de pesquisa de processamento de linguagem natural com objetivo de fornecer aos utilizadores uma interface conveniente e natural para aceder à informação pretendida.

Atualmente, a necessidade de desenvolver sistemas precisos ganha mais importância devido às bases de conhecimento estruturadas disponíveis e à procura contínua para acesso a informação de forma rápida e eficiente.

A resposta às perguntas é uma tarefa complexa que requer uma compreensão do significado de texto de entrada e a capacidade de fundamentar os factos relevantes. Desta forma, estes sistemas são fortemente interligados com o processamento de linguagem, onde é necessário efetuar a análise morfossintática assim como o identificação das entidades mencionadas e conceitos no texto.

Os sistemas de resposta automática a questões são geralmente divididos em 3 tipos, consoante a fonte de dados que as mesmas utilizam, ou seja, dados estruturados, dados semi-estruturados (não possuem uma estrutura definida) e texto livre. Desta forma, a pesquisa nestes sistemas pode ser efetuada sobre uma vasta variedade de informação.

Os primeiros sistemas deste tipo foram desenvolvidas no final dos anos 70 como interface para sistemas de resolução de problemas (por exemplo: Student - resoluções de problemas de álgebra, Lunar - informação sobre as rochas da lua), sendo que eram sistemas restritos ao domínio, enquanto que os sistemas atuais permitem efetuar perguntas em linguagem natural sobre qualquer domínio (Mishra and Jain (2016); Pasca and Harabagiu (2001); Hirschman and Gaizauskas (2001); Kumar et al. (2016); Tahri and Tibermacine (2013)).

2.4.1 Sistema de resposta a questões - abordagem usual

Na figura 2.1 é possível observar o fluxo de um sistema típico de resposta a questões de linguagem natural. Todos os sistemas deste tipo possuem uma análise morfossintática para efetuar a tarefa de tokenização, lematização, atribuição de classes gramaticais aos termos da frase, etc..

A classificação do tipo de pergunta é uma tarefa bastante importante, visto que permite identificar de um modo abrangente a que se refere a pergunta, ou seja, pessoa, localização, animal, etc..

A identificação de entidades e conceitos permite identificar de forma mais precisa sobre o quê a pergunta é efetuada, por exemplo, considerando a seguinte pergunta "*Onde se encontra Aveiro?*", o módulo de classificação da pergunta identifica que o utilizador pretende saber sobre a localização de algo e este módulo identificará que se pretende saber sobre a cidade de Aveiro, visto ser uma entidade mencionada.

Após a identificação das entidades mencionadas e conceitos, é efetuada a extração de informação. Este módulo é chamado de extração devido ao facto de que a maioria dos sistemas deste tipo efetuam a pesquisa numa coleção de texto, extraindo determinadas frases ou parágrafos como resposta à pergunta efetuada, sendo que nem sempre a informação é extraída do texto, mas sim de uma base de dados como *RDF*.

No final é efetuada uma geração de resposta, que segue um determinado padrão, normalmente dependente do tipo de pergunta efetuada.

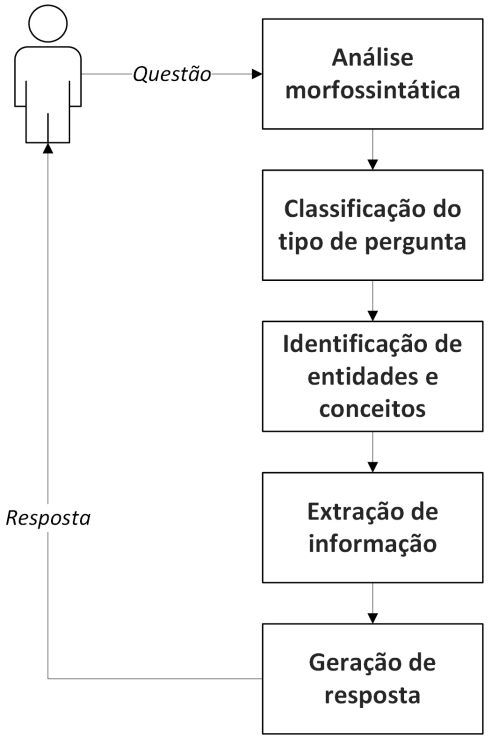


Figura 2.1: Fluxograma de um sistema de resposta automática as questões de linguagem natural

2.4.2 Identificação automática do tipo de pergunta

Foram identificados vários tipos de perguntas, dos quais os mais frequentes são factuais, de listagem e de confirmação. No entanto as perguntas de listagem não são efetuadas da mesma forma que as perguntas factuais, ou seja, enquanto nas perguntas factuais geralmente são utilizados os advérbios "Qu", por exemplo "*Quando nasceu o Albert Einstein?*",

nas perguntas de listagem não são utilizados os advérbios, o que torna difícil identificar o tipo de resposta esperado.

Todos estes tipos devem ser tratados como perguntas. No entanto, isso pode trazer algumas dificuldades para os sistemas que dependem fortemente da identificação de termos como "Quem?", "Onde?", "Quando?" e "O que?" na consulta do utilizador. Por exemplo, identificando a palavra "Quem" na consulta, o sistema sabe que terá que responder sobre alguma entidade, ou a palavra "Onde" indica que a resposta será sobre algum local, etc.

Tendo em conta que as perguntas podem ser feitas em forma de pedidos indiretos ou até comandos, haverá falhas nestes sistemas. Como também não se espera que o utilizador efetue as pesquisas de uma forma estruturada, o que neste caso torna a autossugestão uma boa opção durante a formulação da consulta, como visto no sistema *TR Discover*.

Desta forma, pode concluir-se que existem diferentes tipos de perguntas e várias formas de as classificar, sendo que todos efetuam uma análise da questão, procurando por termos que possam indicar o tipo de pergunta e o tipo resposta esperado e de acordo com Breck et al. (2000), uma boa análise do tipo de resposta esperado, reduz significativamente o número de respostas possíveis.

2.4.3 Criação automática da consulta à base de dados

Como falamos de sistemas de pesquisa semântica, os mesmos utilizam as bases de dados semânticas que podem ser consultadas utilizando uma linguagem específica, ou seja, *SPARQL*, visto ser a linguagem recomendada pela *W3C* para a consulta de *RDF*. Os termos gerados na etapa de desambiguação da consulta passam pelo processo de correspondência entre as propriedades e conceitos ontológicos.

Depois de reunir todas as correspondências ontológicas candidatas que são sintaticamente semelhantes a um termo de consulta, estas são então ordenadas usando dois algoritmos de comparação de *strings* descritos por Winkler (1990); Elbedweihy et al. (2013). O primeiro depende da comparação do número e da ordem dos caracteres comuns, atribuindo uma pontuação elevada aos termos que fazem parte de cada um. Isto é útil uma vez que os conceitos e as propriedades ontológicas são geralmente nomeados dessa forma. Por exemplo, o termo *population* e a propriedade *totalPopulation* recebem uma alta pontuação de similaridade usando este algoritmo (Yao and Van Durme (2014)).

Nesta fase de criação da consulta, a questão do utilizador pode ser interpretada em termos de um conjunto de conceitos, propriedades e instâncias ontológicas que precisam de ser interligados. Como visto anteriormente, os dados semânticos são apresentados em forma de triplos, constituídos por sujeito, predicado e objeto. Desta forma, dado o objeto (a consulta) encontram-se os predicados e sujeitos, onde os predicados apresentam a relação entre objeto e sujeito.

Por exemplo, para a pergunta "*Which television shows were created by Walt Disney?*", é obtido um conjunto de conceitos como "*television show*", "*create*" e "*Walt Disney*" que devolvem os seguintes triplos:

```
?television_show<dbo:creator><res:Walt_Disney>.
```

```
?television_show<dbp:creator><res:Walt_Disney>.
?television_show<dbo:creativeDirector><res:Walt_Disney>.
```

Desta forma, já se torna possível efetuar uma consulta à base de dados semântica, obtendo uma resposta única à pergunta do utilizador.

2.5 Estado da arte

2.5.1 Sistemas representativos

BASEBALL

Um dos sistemas mais conhecidos de resposta a questões de linguagem natural é o BASEBALL (Green Jr et al. (1961); Hirschman and Gaizauskas (2001)), que permitia responder às perguntas sobre os jogos de *basebol* da liga americana ocorridos ao longo de uma temporada. Feita uma pergunta como *"Quantos jogos os Yankees fizeram em julho?"* ou "*Em quantos dias de julho oito equipas jogaram?*", o *BASEBALL* analisava a questão usando o conhecimento linguístico numa forma canónica, que era então usada para gerar uma consulta à base de dados estruturada com dados de *basebol*.

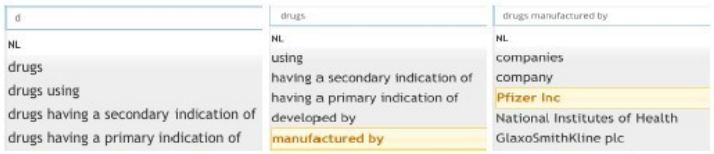
Embora o *BASEBALL* fosse relativamente sofisticado na forma como lidava com a sintaxe e a semântica das perguntas para os padrões dos anos 2000, era limitado em termos do seu domínio (só de *basebol*) e por se destinar a ser essencialmente uma interface para uma estrutura de base de dados e não uma interface para uma grande coleção de texto (Hirschman and Gaizauskas (2001)).

TR Discover

Um dos sistemas mais recentes deste tipo é o *TR Discover*, que além de oferecer a pesquisa utilizando a linguagem natural, ajuda os utilizadores na formulação da pergunta, apresentando a sugestão (Fig. 2.2) durante o preenchimento da consulta ou apresentando uma pesquisa já estruturada após a análise da consulta (Song et al. (2015)).

O *TR Discover* foi desenvolvido para uso futuro com o *Thomson Reuters Cortellis*, que é um produto desenvolvido com base num sistema de dados de domínio farmacêutico (Song et al. (2015)).

Neste sistema o utilizador efetua questões de linguagem natural, que são mapeadas numa linguagem intermediária lógica. A gramática define as opções disponíveis para o utilizador e efetua o mapeamento do inglês para a lógica. O mecanismo de autossugestão (Fig. 2.2) guia o utilizador para questões que são logicamente bem formadas e com maior probabilidade de obter respostas corretas (Song et al. (2015)).



(a) "d" is typed

(b) "drugs" is selected and suggestions are provided

(c) "manufactured by" is picked and "Pfizer Inc" can be chosen to complete a question

Figura 2.2: Autossugestão. (Fonte: Song et al. (2015)).

Sistema de Marius A. Pasca (2001)

O sistema descrito por Pasca and Harabagiu (2001) agrega uma série de requisitos para o sistema a desenvolver, dos quais o mais importante é a taxonomia do tipo de resposta. O objetivo deste sistema difere um pouco do sistema a desenvolver neste estudo, ou seja,uma vez que a resposta à pergunta é pesquisada numa coleção de texto em vez de numa base de dados semântica. No entanto, as técnicas que este sistema aborda são similares ao sistema a desenvolver.

Considera-se que um texto que contém a resposta candidata contém não apenas algumas das palavras-chave da pergunta, mas necessariamente descreve um conceito da mesma categoria semântica que a pergunta efetuada, seja o nome da pessoa, um número, uma data, uma medida, localização ou organização. Desta forma, foi definida a categoria semântica da resposta como o *tipo de resposta esperado*. Para as perguntas de domínio aberto que inquiriram apenas sobre entidades ou eventos, ou alguns dos seus atributos ou funções, como foi o caso das perguntas de teste (*TREC-8* e *TREC-9*), uma taxonomia *off-line* do tipo da resposta esperado pode ser construída, baseando-se em vastos recursos léxico-semânticos como o *WordNet*.

A taxonomia do tipo de resposta neste sistema foi construida utilizando 3 passos. Em primeiro lugar para cada categoria semântica de substantivos ou verbos, foram manualmente examinados os nós conceituais mais representativos e adicionados manualmente como topo da Taxonomia. Além disso, foram adicionadas categorias semânticas abertas correspondentes às entidades mencionadas (Fig. 2.3).

Visto que muitas vezes o tipo de resposta é uma entidade mencionada, é necessário efetuar um mapeamento entre as várias categorias de entidades mencionadas e os topos da Taxonomia (Fig. 2.4).

Por exemplo, numa pergunta de linguagem natural, ao perguntar sobre um valor numerário, a duração ou a velocidade, os mesmos serão mapeados como expressões de quantidade pelo o reconhecedor de entidades mencionadas, enquanto que os conceitos do tipo *Money* são identificados como expressões de dinheiro ou preço pelo reconhecedor de entidades mencionadas. Este mapeamento consiste no segundo passo de criação da taxonomia.

O último passo consiste na ligação manual entre cada elemento do topo da taxonomia e uma ou várias sub-hierarquias do WordNet, o que se pode verificar na figura 2.4, analisando

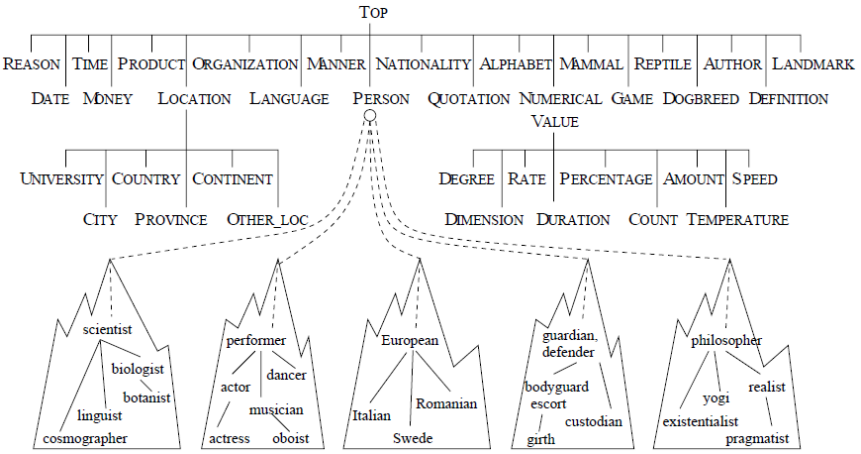


Figura 2.3: Taxonomia do tipo de resposta. (Fonte: Pasca and Harabagiu (2001))

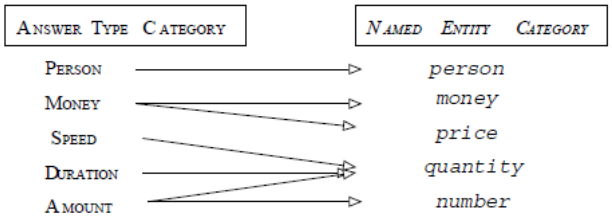


Figura 2.4: Mapeamento. (Fonte: Pasca and Harabagiu (2001))

---

o elemento do tipo *Person*.

A existência de Taxonomia do Tipo da Resposta traz muitas vantagens ao sistema, embora não resolve todos os problemas neste tipo de sistemas. Quando é efetuada uma pergunta em linguagem natural, é necessário identificar palavras que ajudam a identificar o tipo de reposta esperado.

Alguns dos *stems* (termos), quando presentes na pergunta, são unívocos, (por exemplo *who* corresponde sempre a pessoa ou organização). Contudo, maioria dos *stems* são bastante ambíguos (por exemplo *what* que pode perguntar por qualquer coisa). Alguns dos sistemas mais recentes estabelecem uma relação entre o *stem* e a categoria da entidade mencionada. O sistema de Pasca and Harabagiu (2001) é o primeiro a implementar a Taxonomia de Tipo de Resposta abrangente, complementada por um mecanismo robusto de identificação da palavra que determina o tipo de resposta esperado.

Para determinar o tipo de resposta esperado, é efetuada a análise de dependências e posteriormente identificada palavra que tem a dependência da *stem* da pergunta. Para este propósito foi utilizada a própria implementação do analisador *Collins* e foram utilizadas as relações de dependência aprendidas ao treinar o analisador probabilístico. Por exemplo, no caso da pergunta *Q712* da *TREC-9*: *What do tourists visit in Reims?* (Fig. 2.5).

Cada nó terminal na árvore de dependências no nível dois e acima representa um constituinte sintático. Para cada possível constituinte existem regras que identificam o

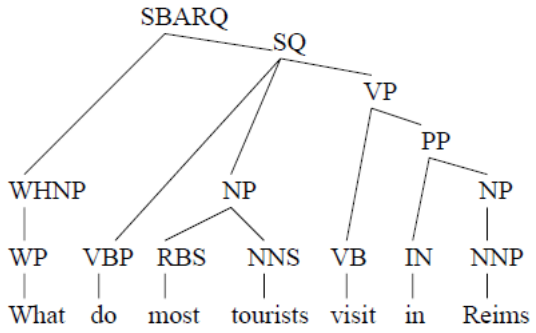


Figura 2.5: Árvore de dependências. (Fonte: Pasca and Harabagiu (2001))

filho principal e a propagação para o seu nó pai. No caso da pergunta *Q712*, a propagação identifica a raiz como *What* que é dependente do verbo *visit*.

Neste caso, o tipo de resposta esperado é um simples objeto do verbo *visit*, mapeado para *Landmark* que está no topo da Taxonomia de Tipo de Resposta. Os conceitos classificados como *Landmarks* são museus, palácios, castelos, catedrais, etc.

A identificação da (s) palavra (s) da pergunta que determina o tipo de resposta baseado em análises de dependências é mais precisa do que métodos empíricos de associar categorias semânticas à raiz da primeira frase ou de palavras-chave, como efetuado nos sistemas IE. A metodologia utilizada neste sistema foi bem sucedida em mais de 90% das perguntas do teste *TREC*, falhando apenas quando a cobertura da taxonomia não era suficiente.

Em geral o tipo de resposta esperado é devolvido como o topo da Taxonomia do Tipo de Resposta. No entanto, existem algumas exceções que foram implementados neste sistema. Uma delas é representada por questões de identificação (por exemplo: *What is platinum?*), cujo tipo de resposta esperado é *DEFINITION*. O reconhecimento de perguntas de definição é baseado na correspondência de um conjunto de padrões sobre a pergunta do utilizador (Fig. 2.6).

(Q-P1): *What {is|are} <phrase_to_define>?*
(Q-P2): *What is the definition of <phrase_to_define>?*
(Q-P3): *Who {is|was|are|were} < person_name(s)>?*

Figura 2.6: Padrões de reconhecimento de perguntas de definição. (Fonte: Pasca and Harabagiu (2001))

O processamento de perguntas que precisam da definição de algo não tem como objetivo encontrar o tipo de resposta esperado, mas sim reconhecer *phrase to define* (entidade a descrever) na pergunta.

K-Extractor

Um outro sistema chamado *K-Extractor*, descrito por Balakrishna et al. (2016), consiste numa interface de linguagem natural de dados estruturados e não estruturados. A linguagem natural é automaticamente convertida em consultas *Sparql* que são executadas sobre a base de dados *RDF*. Ao contrário do sistema a desenvolver, este é dependente do domínio (bactérias e drogas ilícitas).

A extração de conhecimento é efetuada convertendo o conteúdo dos documentos em triplos semânticos. O processamento é iniciado com as tarefas básicas de PLN, como tokenização, *POS-tagging*, desambiguação de palavras, etc. Conceitos e entidades mencionadas são extraídos baseando-se em padrões lexicais e ontologias. No final, relações semânticas, correferências e eventos são extraídos.

As relações semânticas são extraídas em dois níveis. Inicialmente é usado um *parser* semântico *Polaris* que extrai relações semânticas básicas das frases. O sistema a desenvolver seguirá a mesma lógica de extração de conhecimento, sendo que, enquanto este sistema efetua a extração de relações semânticas sobre o texto de entrada para depois converter em triplos, no sistema a desenvolver esta técnica será usada para extrair relações semânticas da pergunta do utilizador para entender a intenção do mesmo, simplificando assim a pesquisa posterior.

O K-Extractor permite aos utilizadores exprimir as suas intenções em Inglês, efetuando as perguntas em linguagem natural que são posteriormente convertidas em consultas Sparql, para a consulta de informação. Em primeiro lugar o texto da pergunta é convertido num grafo semântico, e de seguida o tipo de pergunta é identificado para efetuar as restrições à resposta.

XisQuê

XisQue é um serviço *online* de resposta a perguntas em português de domínio aberto que permite obter a resposta em tempo real, onde as respostas são pesquisadas na coleção dos documentos recuperados da web portuguesa, isto é, a coleção dos documentos escritos em português e disponíveis *online*. Estas respostas consistem na extração de excertos de texto dos documentos recuperados sem nenhum processamento adicional (Branco et al. (2008)).

A infraestrutura do sistema segue o padrão base para os sistemas de resposta automática, ou seja, consiste em 3 fases:

- Processamento da Pergunta: Este fase consiste em 3 passos, identificação do tipo semântico de resposta; identificação de possíveis palavras chave; extração do verbo principal.

- Recuperação de documentos: Neste fase o sistema age como o cliente dos motores de pesquisa (por exemplo: Google, Yahoo), utilizando a lista de palavras chave obtidos na fase anterior, recuperando os documento relevantes.

- Extração da resposta: Esta é a última fase de processamento e consiste em duas tarefas sobre a coleção dos documentos recuperados. As frases que provavelmente contêm uma resposta admissível são selecionadas; Respostas candidatas são extraídas das frases selecionadas; Este sistema devolve até 5 respostas candidatas numa forma sucinta junto com as frases dos quais as mesmas foram extraídas.

O sistema utiliza módulos de processamento de linguagem natural para efetuar segmentação de frases e *tokens*, anotação sintática, análise morfológica, lematização e reconhecimento de entidades mencionadas especialmente elaborados para linguagem portuguesa.

IdSay (I'd Say or I dare Say)

É um sistema de resposta a questões de domínio aberto para português que foi desenvolvido a partir do zero com o objetivo de otimizar espaço e tempo computacional, de modo a que a resposta possa ser rápida. Foi submetido pela primeira vez à *monolingual Portuguese task* da *QA track* do *Cross-Language Evaluation Forum 2008 (QA@CLEF)* (Carvalho et al. (2008)).

O sistema baseia-se nas técnicas de recuperação de informação e, de acordo com o autor, pretende-se ter uma base de recuperação eficiente que possa funcionar de forma independente da língua, podendo ser reutilizada no futuro em outras línguas.

A versão atual do sistema é o mais próximo da pesquisa de palavra-chave simples. A única informação externa que o sistema utiliza para além das coleções de texto é a informação lexical para Português.

IdSay é baseado em técnicas de indexação que foram desenvolvidas a partir do zero utilizando linguagem *C++*. Para este efeito o sistema analisa o texto de entrada em diferentes níveis, construindo um ficheiro índice para cada nível.

- Nível 1 *Document Level*.

  Os documentos são mantidos o mais próximo possível do texto original, além das técnicas de compressão utilizadas. Inclui também tokenização e pré-processamento mínimo para permitir uma recuperação eficiente, ou seja, separação de palavras com espaços e conversão em minúsculas.

- Nível 2 lematização *ou Stemming*

  O sistema utiliza as técnicas de lematização em vez do *stemming* visto que, de acordo com autor, as primeiras apresentam melhores resultados. Os *stop words* não são retirados do texto original.

- Nível 3 Entidades

  Neste nível são pesquisadas todas as sequências de palavras que ocorrem frequentemente nas coleções de texto e se o número de ocorrências for maior do que um determinado limite (100) o sistema considera-os como uma entidade, sendo esta algo significativo como nome de uma organização ou uma série se palavras comuns.

IdSay aceita qualquer pergunta escrita pelo utilizador utilizando interface manual ou um conjunto de perguntas num ficheiro *xml* utilizando interface automática.

Cada questão é analisada pelo módulo de análise de questão para determinar o tipo de pergunta e outras variáveis a serem utilizadas nos módulos e extração e validação de resposta.

Este módulo também determina a *string* de pesquisa e a informação de quais palavras e entidades a serem usados no módulo de recuperação de documentos para produzir uma lista de documentos que correspondem à ambos.

Após a recuperação de documentos são produzidos pequenos segmentos de texto (respostas candidatas) que são passados para o módulo de validação de resposta, que valida respostas e retorna os mais relevantes.

O sistema responde a questões factuais e de definição utilizando o *Wikipedia* como fonte de dados.

Sistema de Paulo Quaresma

O sistema é baseado em duas etapas: para cada pergunta, uma primeira pesquisa seleciona um conjunto de documentos potencialmente relevantes; cada um desses documentos é então analisado para obter uma representação semântica e a resposta para a consulta inicial (Quaresma et al. (2004)).

O sistema precisa de efetuar uma pesquisa preliminar de recuperação de informação, na qual um menor conjunto de documentos potencialmente relevantes é identificado e o principal objetivo do sistema consiste na análise desse conjunto de documentos para obter um representação parcial semântica do seu conteúdo.

Cada consulta é transformada na mesma forma semântica que os documentos recuperados e o processo de inferência tenta obter a resposta para a consulta. Essa abordagem demonstrou muitos problemas de escalabilidade devido ao grande número de documentos e dados associados, levando a uma redução da sua cardinalidade.

Os documentos alvo passam por uma fase de pré-processamento que consiste em duas tarefas essenciais: Interpretação semântica e indexação de informação recuperada. A primeira tarefa permite criar um conjunto de bases de conhecimento, coleção de factos do texto, em que cada base de conhecimento contêm os factos transmitidos por cada texto. A segunda cria os ficheiros que indexam o conjunto completo de documentos com referências à base de conhecimento associada a cada documento.

2.5.2 Análise crítica

Foram analisados 7 sistemas de resposta automática a questões de linguagem natural, desde o mais antigo (Green Jr et al. (1961)) até ao um dos mais recentes (Balakrishna et al. (2016)).

Em geral os sistemas deste género partilham a mesma forma de processamento de linguagem natural, identificação do tipo de pergunta e resposta, e identificação da entidade

foco, enquanto que a forma como a informação para a resposta é obtida e processada vária consoante o sistema.

Para além desta característica, os sistema dividem-se em dois tipos, dependentes do domínio ou não dependentes do domínio, onde os primeiros mostram melhores resultados visto que o sistema já tem padrões predefinidos e até guia o utilizador durante a consulta (Fig. 2.2), como visto no sistema *TR Discover*.

A maioria dos sistemas utiliza as técnicas de extração de informação para obter a resposta à pergunta, retirando as frases do texto original que contêm a entidade foco da pergunta ou que partilham a mesma classe semântica. Alguns dos sistemas identificam as palavras-chave na frase para efetuar a pesquisa normal na Web e extrair a resposta dos documentos recuperados resultantes desta pesquisa.

O sistema apresentado por Quaresma et al. (2004) efetua muito processamento tanto sobre a coleção dos documentos recuperados como na criação de uma ontologia complexa. De todos os sistema vistos, este é o que efetua mais processamento, obtendo resultados piores, em comparação entre os sistemas de domínio aberto. É possível distinguir entre sistemas independentes de domínio e sistemas específicos de domínio (como sistemas de ajuda) (Hirschman and Gaizauskas (2001)).

Os sistemas *BASEBALL* e *TR Discover* são exemplos de sistemas dependentes de domínio, mas, apesar disso, foram tidos em conta uma vez que utilizam as técnicas de processamento de linguagem natural que serão abordadas neste estudo.

2.6 Conclusão

Com a pesquisa efetuada torna-se claro a existência e o crescimento da necessidade de sistemas de pesquisas semânticas com a utilização de linguagem natural. Apesar de atualmente termos bons sistemas de pesquisas, os mesmos retornam uma lista de resultados em vez de uma resposta sucinta, obrigando o utilizador a filtrar os resultados até encontrar a informação pretendida, sendo por isso essencial o desenvolvimento de um novo sistema que tenha em conta esta questão.

Foram apresentadas algumas das aplicações de pesquisa semântica que permitem a pesquisa utilizando a linguagem natural, que podem ser específicos de domínio ou não, onde os primeiros são mais fáceis em tratar as perguntas do utilizador porque a diversidade dos termos de consulta é reduzida, enquanto que os segundos precisam de efetuar o processo de desambiguação da consulta, conseguindo uma melhor correspondência entre a consulta do utilizador e a ontologia.

A maior dificuldade no desenvolvimento de sistemas deste tipo é efetuar uma boa desambiguação e encontrar as correspondências dos termos resultantes com a ontologia.

Todos estes aspetos foram tidos em conta durante o desenvolvimento do sistema proposto, com o objetivo de criar uma aplicação simples e poderosa para responder a questões do utilizador.

Sistema	Método de obtenção da resposta	Tarefas PLN	Elementos extraídos da pergunta	Fonte de informação	Domínio aberto
BASEBALL (Green Jr et al. (1961))	Pesquisa no dicionário	Análise sintática Análise do conteúdo	Verbos Substantivos	Dicionário	Não
TR DISCOVER (Song et al. (2015))	Consultas Sparql	Tokenização Lematização Análise lexical	Verbos Substantivos	Jena TDB triple store	Não
Sistema de Marius A. Pasca (Pasca and Harabagiu (2001))	Pesquisa numa coleção de texto	Tokenização Lematização Árvore de dependências	Entidade mencionadas Conceitos	WordNet	Sim
K-Extractor (Balakrishna et al. (2016))	Extração de informação dos múltiplos recursos de texto	Tokenização Análise morfossintática Desambiguação do sentido da palavra	Conceitos Entidades mencionadas Relações semânticas Eventos	RDF Triple store	Não
XisQuê (Branco et al. (2008))	Extração de uma coleção de texto sem processamento adicional	Sentence segmentation Token segmentation Análise sintática Análise morfológica Lematização Identificação de Entidades mencionadas	Token para identificação do tipo de pergunta Keywords relevantes Verbo principal	Documentos recuperados	Sim
IdSay (Carvalho et al. (2008))	Pesquisa de perguntas na base de conhecimento que em maioria dos casos consiste em textos de linguagem natural.	Tokenização Separation ow word with spaces Lowercase conversion Lematização	Frequent words	Coleção de texto	Sim
Sistema de Paulo Quaresma (Quaresma et al. (2004))	Extração de frases que partilham o mesmo facto da pergunta.	Análise sintática Interpretação semântica	Factos	Documentos	Sim

Tabela 2.2: Comparação de sistemas de resposta automática às questões.

Capítulo 3

Recursos e Ferramentas utilizados

3.1 Introdução

Para o desenvolvimento das duas variantes do sistema foram utilizadas ferramentas e recursos de diferentes áreas, tais como Processamento de Linguagem Natural e Web Semântica.

O objetivo deste capitulo é apresentar estas ferramentas de uma forma resumida, mostrando as vantagens e características importantes.

O conteúdo é dividido em 3 secções principais: ferramentas de Análise de Linguagem Natural, Fontes de informação semântica e Bases de Dados semânticos.

3.2 Ferramentas de análise de linguagem natural

Para o processamento de linguagem natural foram selecionadas duas ferramentas, *Freeling* e *Maltparser*. Na pesquisa efetuada foi identificado que a maioria dos sistemas de resposta automática são desenvolvidos para linguagem inglesa, sendo que as ferramentas de processamento de linguagem natural por eles utilizados são mais otimizados para esta linguagem e não são ideais para português. Desta forma, nesta fase o objetivo consistiu na pesquisa de ferramentas que são capazes de efetuar uma boa análise de linguagem natural portuguesa.

A ferramenta *Freeling* foi escolhida como opção para efetuar toda a análise morfossintática das frases de entrada fornecendo todo o tipo de análise base que necessitamos, como também permite fácil configuração e gestão, permitindo introduzir novas regras de análise e o *Maltparser* essencialmente para a análise de dependências, permitindo a identificação da entidade foco da frase.

3.2.1 *Freeling*

O *Freeling* é uma biblioteca *open-source* de processamento de texto que oferece uma ampla gama de funcionalidades de análise para várias línguas. O projeto *FreeLing* foi re-

alizado no centro de pesquisa *TALP2* para fornecer avanços na disponibilidade geral de ferramentas e recursos básicos de processamento de linguagem natural. Essa disponibilidade deve permitir avanços mais rápidos em projetos de pesquisa e menores custos no desenvolvimento industrial de aplicações de processamento de linguagem natural. O projeto é distribuído como uma biblioteca que pode ser chamada a partir de uma aplicação que necessite de serviços de análise (Padró and Stanilovsky (2012)).

O *Freeling* fornece vários módulos de processamento de linguagem natural, sendo que foram utilizados os seguintes:

- Identificador da linguagem
- Tokenização
- Separador de frases
- Analisador morfossintático

Tokenizador - É o primeiro módulo no processamento do texto, que converte o texto num vetor de objetos do tipo *word* de acordo com as suas expressões regulares.

Separador de frases - recebe a lista de objetos do tipo *word* produzidos pelo módulo anterior, e agrupa-os em frases (*sentences*) quando é encontrado o limite da frase e retorna a lista de frases. Tendo em conta que o sistema desenvolvido recebe uma pergunta do utilizador, esta lista devolverá sempre uma frase. De modo a tornar o módulo *thread-safe*, o módulo mantém diferentes *buffers* internos para diferentes *threads*, assim a função que chama o seguinte módulo deve abrir uma sessão com identificador único para aceder ao seu *buffer* interno.

Análise morfossintática - este módulo não efetua nenhum processamento, mas sim simplifica a instanciação de sub-módulos que efetuam análise morfológica. Na fase de instanciação deste módulo é passado como parâmetro o objeto *maco_options*, que indica quais os sub-módulos têm de ser criados e quais os ficheiros serão usados para a criação dos mesmos. Quando o módulo recebe um objeto do tipo *sentence*, a frase é automaticamente passada para todos os sub-módulos ativos e devolve o resultado final.

O módulo de Análise Morfológica permitiu usar os seguintes sub-módulos:

Detetor de pontuação - atribui classe sintática a símbolos de pontuação.

Detetor de números - é dependente da linguagem e tem como objetivo detetar símbolos numéricos como 1,220.54 ou duzentos e sessenta e cinco, atribuindo um valor normalizado como lema.

Detetor de datas - como o módulo de deteção de números, é um conjunto de *Augmented Transition Networks*, específicas de idioma. Para os idiomas que não possuem uma *ATN* específica, é efetuada uma análise padrão que deteta padrões de datas simples (e.g. DD-MM-AAAA, MM / DD / AAAA, etc)

Pesquisa no dicionário - este módulo tem duas funções, procura a forma da palavra no dicionário com objetivo de descobrir o lema e classe sintática e aplica as regras de afixação ou composição para conseguir os mesmos resultados em caso de forma da palavra for derivada e não está incluída no dicionário, como por exemplo palavras com sufixo, como emoção, amplitude, presença, etc..

Análise de dependências - permite obter a árvore de dependências entre as palavras na questão do utilizador, sendo um passo importante na identificação do foco da frase.

Reconhecimento de entidades mencionas - permite identificar entidades mencionadas na pergunta do utilizador e simplificar a sua identificação no mundo semântico.

3.2.2 Maltparser

O *MaltParser* é um *parser* sintático baseado em dependências e um sistema de análise de dependências orientado por dados, que pode ser usado para induzir um modelo de análise a partir de um *treebank* e analisar novos dados usando um modelo induzido. Suporta vários algoritmos de análise e de aprendizagem e permite o uso de modelos definidos pelo utilizador, que consistem em combinações arbitrárias de recursos lexicais, características sintáticas e estrutura de dependências (Nivre et al. (2006)).

O *MaltParser* é uma implementação de análise de dependência indutiva, onde a análise sintática de uma frase equivale à derivação de uma estrutura de dependências e onde a aprendizagem pela máquina indutiva é usada para orientar o analisador em pontos de escolha não deterministas. Esta metodologia de análise é baseada em três componentes essenciais:

- Algoritmos de análise determinística para a construção de gráficos de dependência
- Modelos de características baseados na história para prever a próxima ação do analisador
- Aprendizagem da máquina discriminante para mapear as histórias para as ações do analisador

Dadas as restrições impostas por esses componentes, o *MaltParser* foi projetado para oferecer a máxima flexibilidade na forma como os componentes podem ser variar independentemente um do outro (Lavelli et al. (2009)).

Os recursos são definidos em termos de forma de palavra (LEX), classe sintática (POS) ou tipo de dependência (DEP) de um *token* definido em relação a uma das estruturas de

dados *STACK*, *INPUT* e *CONTEXT*, usando as funções auxiliares *HEAD, LC, RC, LS* e *RS* (Lavelli et al. (2009)).

3.3 Fontes de informação semântica

3.3.1 *BabelNet*

Com o objetivo de centralizar a fonte de informação é utilizado o *BabelNet* que segue a estrutura de um dicionário tradicional e, consequentemente, consiste num grafo onde os nós representam conceitos e entidades mencionadas e os ramos expressam relações semânticas entre eles. Os conceitos e as relações são recolhidos a partir da maior base de dados léxica, WordNet (Fellbaum (1998)) e a partir da Wikipedia.

3.3.2 *DBpedia*

DBpedia é uma comunidade que tem como objetivo extrair a informação estruturada de *Wikipedia* e torná-la disponível na Web. *DBpedia* permite efetuar consultas sofisticadas à informação do *Wikipedia*. De acordo com *DBpedia*, atualmente a maioria das bases de conhecimento cobrem apenas domínios específicos, que são criados por grupos relativamente pequenos, e em poucos casos se mantêm atualizadas. Desta forma, *DBpedia*, utiliza a fonte de conhecimento de *Wikipedia*, que se tornou uma das principais fontes de conhecimentos na Internet, editada por milhares de utilizadores.

A versão em inglês da base de dados *DBpedia* descreve 4,58 milhões de coisas, dos quais 4,22 milhões são classificados numa ontologia consistente, incluindo 1.445.000 pessoas, 735.000 lugares (incluindo 478.000 lugares povoados), 411.000 obras criativas (incluindo 123.000 álbuns de música, 87.000 filmes e 19.000 jogos de vídeo), 241.000 organizações (incluindo 58.000 empresas e 49.000 instituições educacionais), 251.000 espécies e 6.000 doenças.

3.4 Jena

3.4.1 *Apache Jena Triple Store*

O *TDB* é um componente do Jena para armazenamento e consulta *RDF*. O *TDB* pode ser usado como um armazenamento *RDF* de alta performance numa única máquina, podendo ser acedida e gerida por linha de comandos e através da *API Jena*.

3.5 Corpora usado

O corpora usado consiste num conjunto de dados do *Bosque corpus* da Floresta Sintática, que por sua vez é um *treebank* publicamente disponível para português, criado como um projeto de colaboração entre o projeto *VISL* e *Linguateca*, e baseia-se na revisão humana da produção do analisador *PALAVRAS*.

O projeto Floresta Sintática inclui três corporas, Floresta Virgem, Bosque e Selva, entre os quais, o do Bosque foi utilizado e que consiste num subconjunto de Floresta, totalmente revisto pela equipa linguística, com aproximadamente 162,484 unidades lexicais.

Capítulo 4

Framework para Resposta Automática a Questões (em Português)

Este capitulo visa apresentar e descrever a arquitetura geral do sistema de resposta automática a perguntas efetuadas em linguagem natural. Serão abordadas as diferentes fases de análise e processamento de informação que permitem responder à questão do utilizador.

4.1 Arquitetura

O sistema desenvolvido consiste em 6 módulos distintos interligados entre si, permitindo receber a pergunta do utilizador, processá-la e devolver uma possível resposta à mesma (Fig. 4.1).

É possível observar (Fig. 4.1) a existência de uma área onde toda a informação derivada do processamento da pergunta é armazenada. Para além da área de informação derivada, existe a base de dados do sistema onde toda a informação consultada de recursos online é armazenada. Desta forma, o processo de resposta à questão de linguagem natural consiste em três principais fases de processamento, duas fases de extração de informação dos recursos online apropriados e a fase final, que agrega toda a informação derivada da análise de linguagem natural e informação consultada, devolvendo a resposta à questão do utilizador.

Para melhor compreensão dos módulos que constituem o sistema e seus dados de entrada e saída é apresentado o fluxo de processamento de uma possível pergunta factual "*Onde nasceu o Albert Einstein?*"(Fig. 4.2). Esta é uma das perguntas típicas que o sistema é destinado a analisar e devolver uma possível resposta ao utilizador. O processamento é dividido em 6 passos fundamentais, onde é possível ver o resultado de cada um. Os posteriores exemplos durante a descrição dos módulos e os seus algoritmos de processamento terão como referência a pergunta apresentada na figura 4.2.

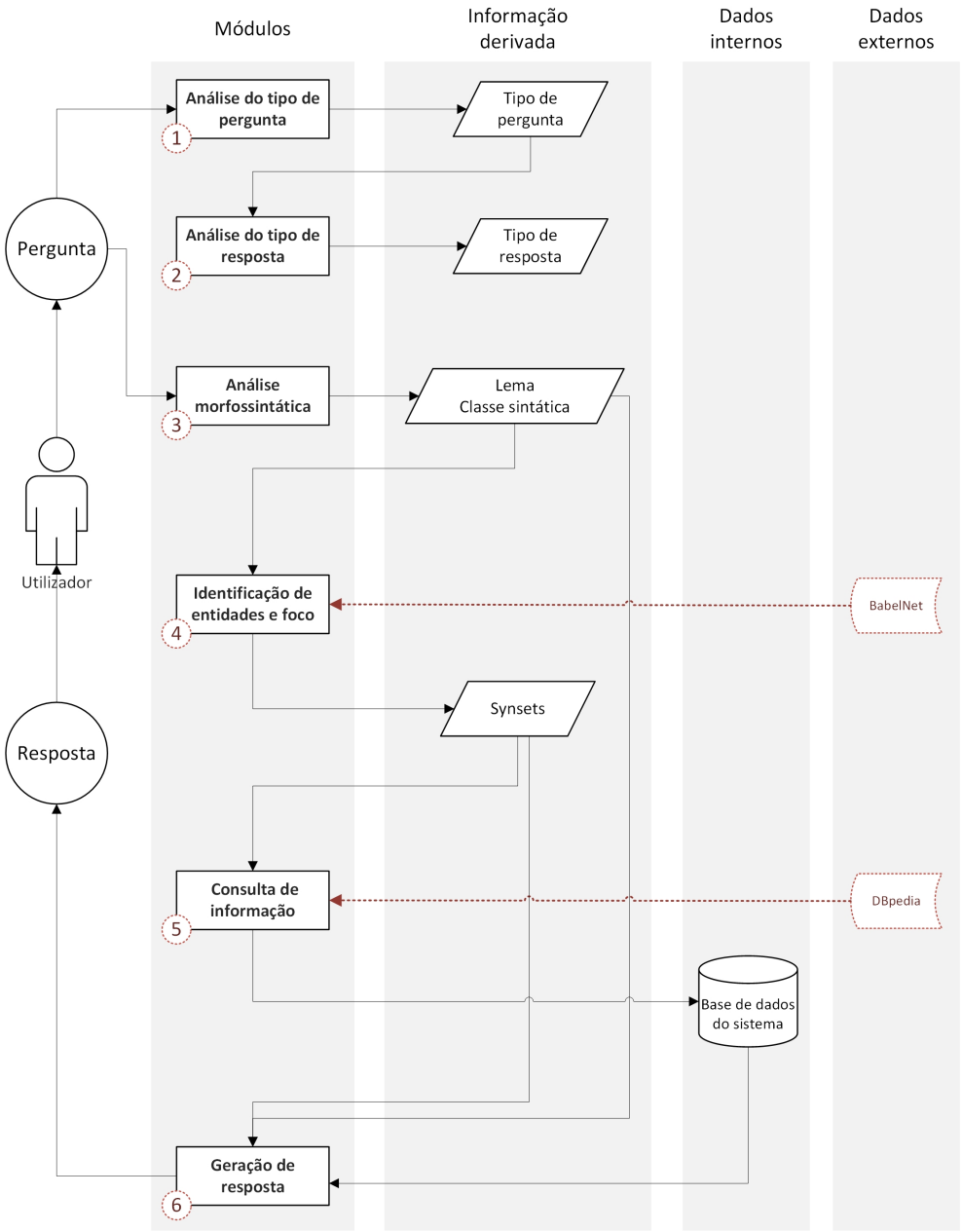


Figura 4.1: Arquitetura funcional do sistema

4.1.1 Análise do tipo de pergunta

O primeiro módulo no processamento da questão do utilizador consiste em identificar, de uma forma generalizada, o assunto da questão do utilizador.

Como foi referido anteriormente, existem diferentes tipos de perguntas que podem ser efetuadas nos sistemas deste tipo, sendo que o sistema desenvolvido tratará perguntas de 3 tipos, nomeadamente, causais, de listagem e definição.

Desta forma, foram analisados alguns sistemas já existentes e identificou-se que a classificação do tipo de pergunta pode ser efetuada de duas formas: tipo de pergunta que depende de seguimento de um padrão ou não.

Os sistemas que usam um padrão para identificação do tipo de pergunta classificam o tipo de pergunta identificando palavras como *Quem? Quando? Como? Porquê? Quando?* na pergunta do utilizador e derivam informação a partir das mesmas.

Os sistemas que não seguem o padrão, usam outras palavras da frase para identificar o tipo de pergunta, sendo este processamento mais complexo e destinado mais a sistemas que têm o domínio predefinido.

Desta forma, este módulo é o primeiro na identificação da intenção do utilizador, que permitirá posteriormente identificar a real intenção do mesmo e conseguir efetuar a pesquisa da resposta de uma forma mais eficiente.

4.1.2 Análise do tipo de resposta

Este módulo tem como objetivo identificar o tipo de resposta esperado tendo em conta o tipo de pergunta identificado. Desta forma, este módulo está diretamente ligado com o módulo anterior. O tipo de resposta é expressado em forma de classe semântica ou tipo de dados *xml*, visto que esta informação será usada na construção da consulta à base de dados do sistema. O estudo de Pasca and Harabagiu (2001) usou a taxonomia do tipo de resposta para identificar os possíveis tipos, que foram derivados do *WordNet*.

Para efetuar a agregação de informação é efetuada uma consulta à base de dados externa (*DBpedia*), onde a consulta é constituída também pelo tipo de resposta esperado. Desta forma, o tipo de resposta esperado é uma classe de ontologia ou um tipo de dados *xml*. Na base de dados semântica os objetos são descritos e interligados entre si por uma ontologia que descreve o significado das coisas. Desta forma, foi analisada a ontologia da base de dados semântica que será consultada, e identificaram-se diferentes classes e tipo de dados.

4.1.3 Análise morfossintática

Este módulo tem como objetivo efetuar toda a análise morfossintática ao texto de entrada, que difere em diferentes sistemas. A maioria dos sistemas têm como texto de entrada uma coleção de textos ou, como no caso deste sistema, a pergunta do utilizador. O módulo efetua o processamento da pergunta do utilizador, dividindo-a em *tokens*, e identifica a qual classe sintática cada *token* pertence e qual é o seu lema. Uma das maiores vantagens deste módulo é a construção de árvore de dependências que permite entender a relação entre diferentes termos da frase, que por sua vez permitirá identificar entidades foco da frase, efetuando mais um passo na identificação da intenção do utilizador.

4.1.4 Identificação de entidades e foco

Este módulo é responsável pela identificação e consulta de *synsets* para cada um dos *tokens*. Esta prática é muito comum nos sistemas deste tipo, visto que é necessário iden-

tificar o significado da palavra para que a mesma possa ser entendida da mesma forma no mundo semântico. *Synsets* são considerados sinónimos, e podem ser extraídos a partir de uma base de dados lexical como *WordNet*, que é a mais comum em sistemas de reconhecimento de linguagem natural inglesa. Desta forma, tendo uma palavra, é possível identificar um conjunto de *synsets*, expandindo assim o significado da mesma, conseguindo assim melhores resultados no processamento posterior. Para além da extração dos sinónimos, é efetuada a identificação das entidades mencionadas e o foco da pergunta. Como já referido anteriormente, as entidades podem ser de dois tipos, mencionadas (pessoas, organizações, cidades, etc.) e conceitos (por exemplo luz, agua, estrela, etc.). Desta forma, é feita uma análise e são identificadas diferentes entidades, sobre as quais o utilizador pretende saber algo, sendo o último passo na identificação da intenção do utilizador.

4.1.5 Consulta à fonte de informação

A consulta à fonte de informação é o penúltimo módulo na fase de análise da questão do utilizador, e efetua a agregação de toda a informação necessária sobre as entidades identificadas anteriormente. A agregação de informação é efetuada construindo e executando as consultas à base de dados externa, (*Wikipedia*, *DBpedia*, etc.).que são escritas em linguagem *Sparql*, executadas posteriormente no respetivo *endpoint*. Desta forma, o objetivo deste módulo consiste em obter a informação complementar que permite responder à questão efetuada.

4.1.6 Geração de resposta

O último módulo no processamento da questão do utilizador consiste em efetuar pedidos à base de dados do sistema com objetivo de conseguir a resposta e apresentá-la ao utilizador. Aqui são construidas as consultas finais à base de dados do sistema, onde está armazenada a informação sobre as entidades extraídas anteriormente. Após a extração de um conjunto de possíveis respostas, as mesmas são apresentadas ao utilizador.

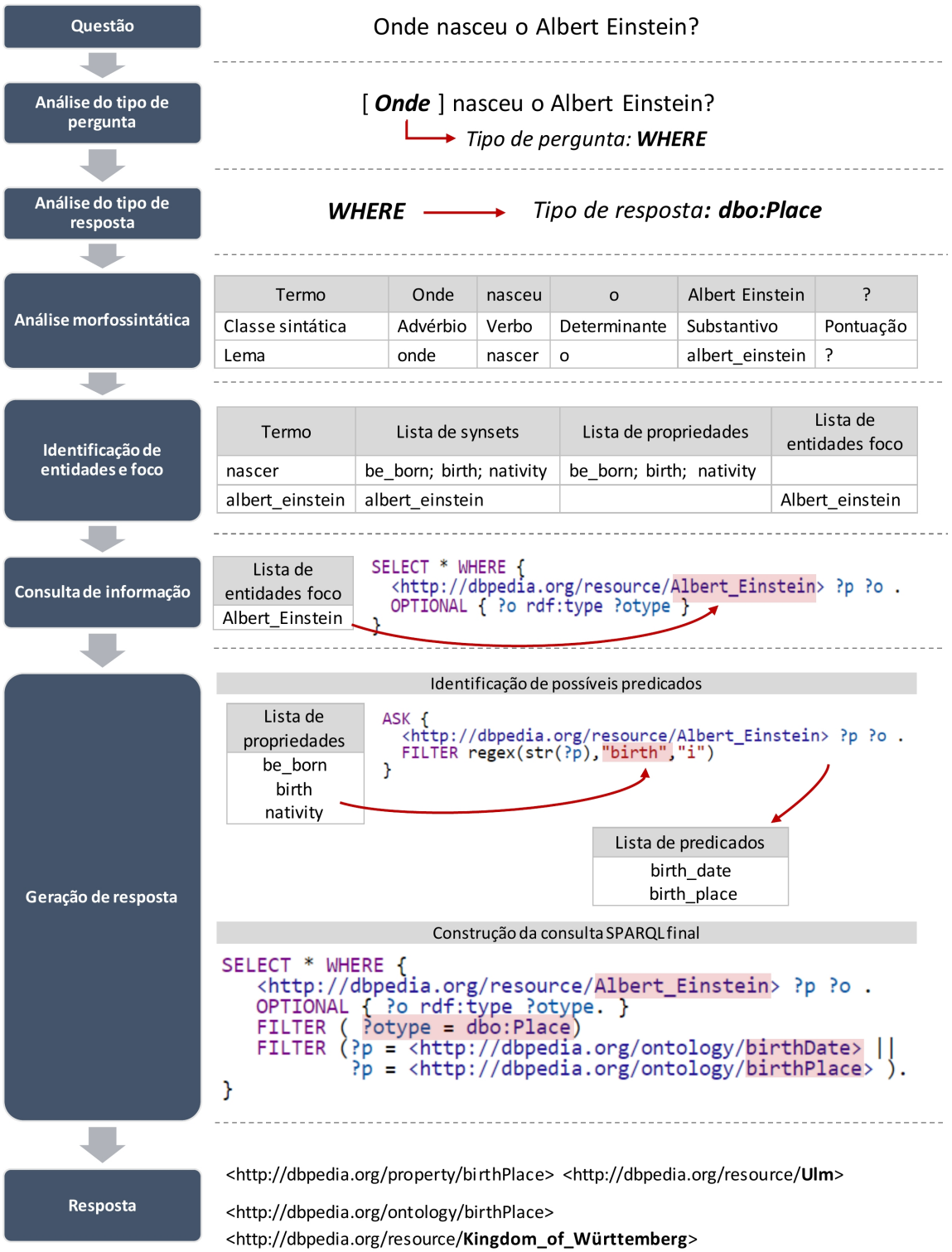


Figura 4.2: Exemplo de análise de uma possível pergunta

Capítulo 5

Variantes do Sistema e Resultados da sua Avaliação

Neste capítulo serão descritas duas variantes do sistema, descrevendo diferentes módulos que o constituem e por fim são apresentados os métodos de avaliação junto com os respetivos resultados.

5.1 Variante 1

5.1.0.1 Análise do tipo de pergunta

O processamento deste módulo consiste em identificar a ocorrência de palavras com o objetivos de identificar o tipo de pergunta (Fig. 5.1).

5.1.0.2 Análise do tipo de resposta

O tipo de resposta está diretamente relacionado com o tipo de pergunta e consiste num conjunto de classes e tipos de dados (Fig. 5.2) que são utilizados na base de dados externa. A figura 5.1 representa o tipo de resposta consoante o tipo de pergunta. Na terceira coluna é possível observar o tipo de resposta esperado, que pode ser tanto um conjunto de classes como um conjunto de tipo de dados. Tendo em conta que a consulta da informação será efetuada à base de dados do DBpedia, as classes apresentadas e os tipos de dados foram consultadas na mesma. Na DBpedia existe um conjunto de classes à qual um determinado objeto pertence. Estas classes estão organizadas numa hierarquia (Fig. 5.2), e uma vez que um objeto pertence a uma determinada classe, o mesmo também pertence à classe pai.

Desta forma, quando o tipo de resposta consiste numa determinada classe, esta classe é sempre a superclasse a seguir a classe *Thing*. Por exemplo, se o tipo de pergunta for *Who* o tipo de resposta será *Person* que corresponde a uma pessoa ou *Agent* que corresponde a uma organização. Na figura 5.2 é possível observar a ontologia do DBpedia e algumas das classes principais, e ainda, observar que a classe *Employer* e *Organisation* têm a mesma classe pai *Agent*, o que permite facilitar a filtragem na fase de construção da resposta final.

Termo	Tipo de pergunta	Tipo de resposta
Quem	WHO	dbo:Person dbo:Agent
Quando	WHEN	xsd:date xsd:integer
Onde	WHERE	dbo:Place xsd:float
Qual	WHICH	dbo:Person dbo:Place dbo:Agent dbo:Game dbo:Dog_breeds xsd:date xsd:time xsd:integer xsd:float xsd:long xsd:double
Quanto	HOWMUCH	xsd:time xsd:integer xsd:float xsd:long xsd:double xsd:duration xsd:positiveinteger xsd:negativeinteger xsd:unsignedint xsd:unsignedlong xsd:unsignedshort
Quem / O que é / era / foi	DEFINITION	dbo:abstract

Figura 5.1: Mapeamento entre o tipo de pergunta e o tipo de resposta

5.1.0.3 Análise morfossintática

Para a análise morfossintática da pergunta do utilizador foi utilizada a ferramenta *Freeling* descrita anteriormente. Tendo a pergunta do utilizador como o conteúdo de entrada, é obtida uma lista de *tokens*, com o respetiva lema e classe gramatical, como também uma árvore de dependências entre as mesmas na frase.

O resultado deste módulo é um conjunto de *tokens* com o seu lema e classe sintática. Na figura 5.3 é possível observar os dados de entrada e de saída deste módulo.

Figura 5.2: Estrutura de classes da ontologia da DBpedia. (Fonte: http://mappings.dbpedia.org/server/ontology/classes/)

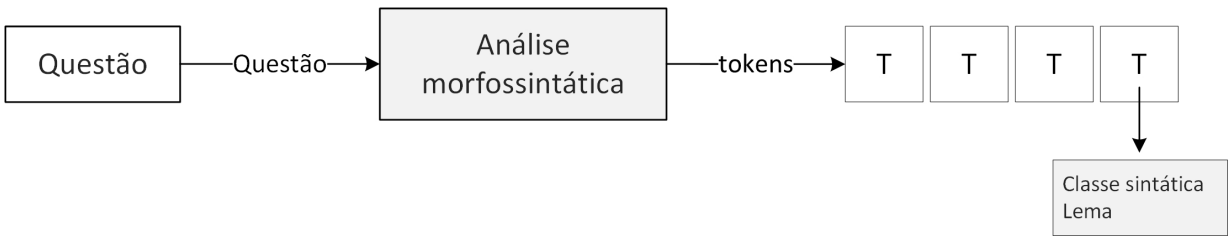


Figura 5.3: Dados de entrada e saída do módulo de análise morfossintática

5.1.0.4 Identificação de entidades e foco

Como já referido, este módulo identifica e consulta os *synsets* para cada um dos *tokens* da frase. Para este efeito foi utilizado o *BabelNet*, sendo um dicionário multilingue e uma rede semântica que interliga os conceitos e entidades mencionadas numa rede de relações semânticas, chamados *BabelSynsets*. *BabelNet* agrega diferentes fontes de informação que podem ser selecionadas na fase de consulta. Esta possibilidade é importante visto que cada fonte é destinada para o tipo de informação diferente. Desta forma, foi feita a seleção de fontes consoante a classe de palavra à qual um determinado *token* pertence (Fig. 5.4).

Tendo em conta alguns testes efetuados, concluiu-se, que entre todas as possíveis fontes fornecidas por *BabelNet*, a fonte *OMWN* (*Open Multilingual WordNet*) fornece o melhor resultado na consulta de synsets para os verbos e adjetivos, enquanto as restantes fontes

Classe sintática	Fonte
Substantivo	• Wikipedia • WordNet • Wikidata • OmegaWiki • Wikipedia redirection • Wiktionary • Automatic translation of a Wikipedia concept
Verbo	Open Multilingual WordNet
Adjetivo	

Figura 5.4: Fonte de dados de consulta de *BabelSynsets* consoante a classe sintática

fornecem bons resultados para os substantivos.

É possível notar (Fig. 5.4) que para os substantivos é usado um conjunto de fontes, visto que durante os testes efetuados algumas das fontes não apresentavam nenhum resultado, e ao consultar múltiplas fontes é garantido que pelo menos uma das fontes devlve resultado pretendido, enquanto na consulta de *synsets* para os verbos e adjetivos não se verificaram falhas deste género.

Pretende-se assim consultar *synsets* para todos os *tokens* utilizando diferentes fontes, enriquecendo assim cada *token* com um conjunto de sinónimos, para além do seu lema e a classe sintática.

Como as perguntas são efetuadas em linguagem natural portuguesa não é utilizado o *WordNet* tradicional como nos sistemas deste tipo destinados para inglês, mas sim *WordNet multilingue*, que nos permite obter os sinónimos em diferentes línguas para um determinado *token*. Apesar da linguagem de entrada ser em português, a sua análise tem de ser efetuada obrigatoriamente em inglês, uma vez que as relações semânticas na base de dados externa são descritas em inglês. Desta forma, não só obtemos os *synsets* para cada *token*, mas também a tradução do mesmo para a língua pretendida, inglês.

A consulta de *synsets* pode ser expandida de forma a enriquecer a informação que será usada na pesquisa da resposta final. Isso é feito de seguinte forma: Para cada *token* é consultado um conjunto de *synsets*; Para cada *synset* obtido, são consultados os hiperónimos, que depois são adicionados à mesma lista dos *synsets* do *token*. Os hiperónimos permitem generalizar os conceitos, abrangido assim a maior área de conhecimento. Na figura 5.5 é possível observar o processo de identificação e consulta de *synsets* para cada *token*. Após a consulta obtém-se uma lista de *tokens*, com a informação sobre o seu lema, classe sintática e um conjunto de *synsets*, onde cada um contém um identificador que o caracteriza indicando se o mesmo é uma entidade mencionada, um conceito ou nada. Esta informação é utilizada na identificação do foco da frase, com objetivo de entender sobre o quê ou ou quem o utilizador pretende obter resposta.

Para identificar o possível foco da pergunta, é efetuada uma filtragem pela classe sintática do *token* e o tipo de *synset*, ou seja, se o *token* for substantivo e o tipo de *synset* for entidade mencionada ou conceito, o *synset* é adicionado à lista de possíveis focos da

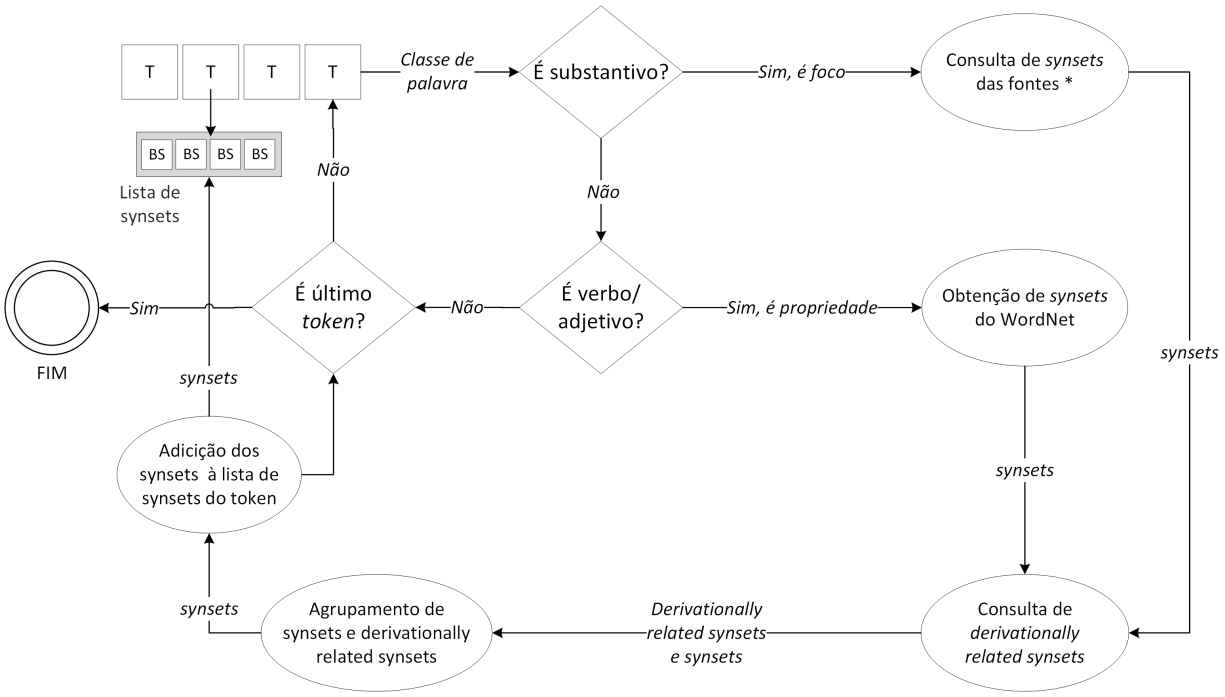


Figura 5.5: Processo de obtenção de *synsets* para cada *token*
* - (Fig. 5.4)

pergunta. Os *synsets* de um *token* não substantivo são considerados como propriedades dos substantivos (propriedades da entidade foco), sendo colocados na respetiva lista de propriedades. Na figura 5.6 é possível observar os dados de entrada e de saída do módulo de identificação de entidades e foco, e o seu processo pormenorizado na figura 5.7.

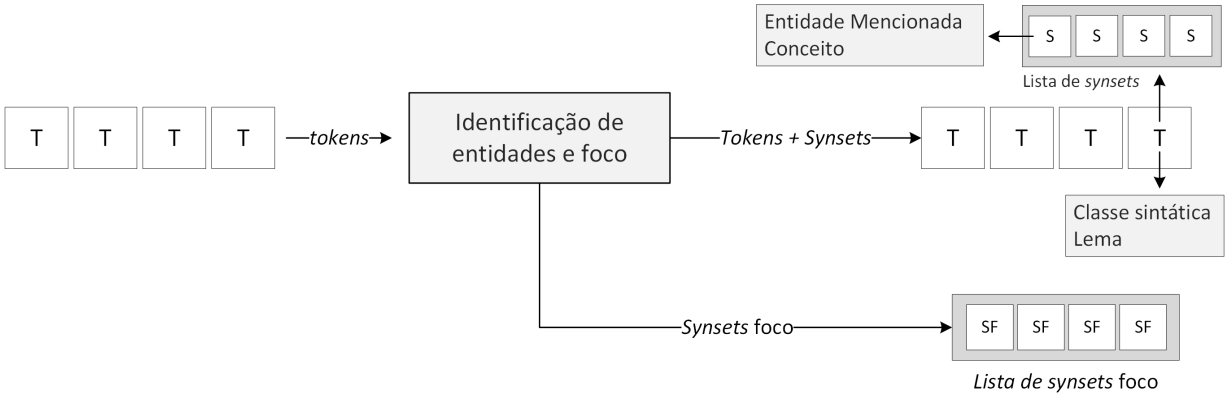


Figura 5.6: Dados de entrada e saída do módulo de identificação de entidades e foco

Resumidamente, o objetivo deste módulo consiste na consulta de sinónimos para cada um dos *tokens*, e a posterior filtragem para a identificação das possíveis entidades foco da pergunta e as suas propriedades.

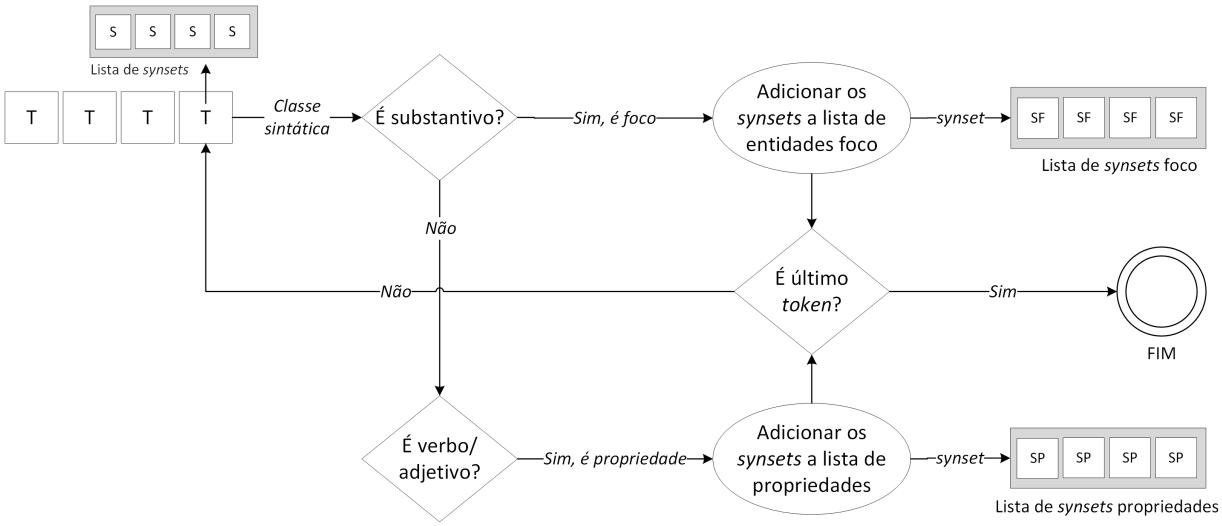


Figura 5.7: Processo de identificação de entidades foco e suas propriedades

5.1.0.5 Consulta de informação

Este módulo efetua a consulta de toda a informação necessária que permite responder à pergunta sobre as entidades foco identificados anteriormente. A consulta é efetuada à base de dados semântica *DBpedia* utilizando o seu *Sparql Endpoint*. Percorrendo a lista de possíveis focos da pergunta, é gerada uma consulta *Sparql* e é efetuada a consulta à base de dados externa (Fig. 5.8).

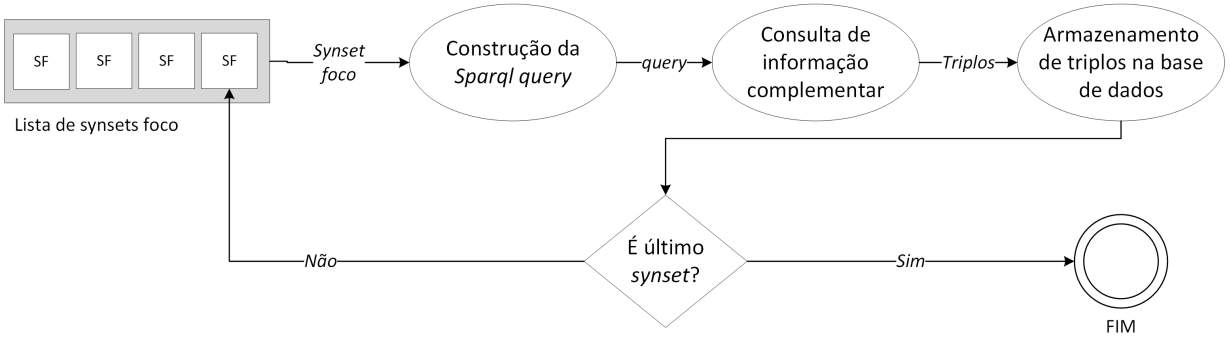


Figura 5.8: O processo de consulta de informação complementar sobre a entidade foco

Uma das tarefas importantes durante a consulta de informação consiste em entender que tipo de informação será consultado, ou seja, à qual nível de expansão é necessário chegar.

A consulta à base de dados semântica resulta num conjunto de triplos, constituídos por sujeito, predicado e objeto. Desta forma, quando fala-se em nível de expansão, tem-se em conta identificar o que para além do sujeito, predicado e objeto será consultado.

Em primeiro lugar é necessário de consultar todos os triplos que têm como sujeito a

entidade foco da questão. Este conjunto de triplos permite obter propriedades da entidade foco (por exemplo: se a entidade foco é uma pessoa, obtemos propriedades como altura, peso, nacionalidade, etc.). Este tipo de informação permite obter uma possível resposta à questão.

Contudo, este conjunto de triplos não é suficiente para a resposta à questão, porque não permite filtrar o resultado por tipo de resposta esperado. Desta forma, para conseguir efetuar a filtragem por tipo de resposta é necessário de conhecer o tipo de dados ou a classe semântica do objeto.

Tendo em conta a figura 4.2, temos uma questão "Onde nasceu o Albert Einstein?". Identificou-se a entidade foco *Albert_Einstein* para qual foi consultado um conjunto de triplos que possuem características sobre esta entidade, como data de nascimento, local de nascimento, local de residência, etc. (Fig. 5.9). Estas propriedades interligam o sujeito (Albert_Einstein) com os objetos, que possuem um determinado tipo, sendo este um tipo de dados ou a classe semântica, que se pretende conhecer para efetuar a filtragem por tipo de resposta esperado.

Assim, com a consulta pretende-se obter um conjunto de triplos que possuem as propriedades da entidade foco sobre quais o utilizador pretende obter resposta e o tipo do objeto que corresponde ao tipo de resposta esperado (Fig. 5.9).

:Albert_Einstein	dbpedia:ontology/knownFor	:Special_relativity	owl:sameAs	<http://eu.dbpedia.org/resource/Erlatibitate_berezia>
:Albert_Einstein	dbpedia:ontology/citizenship	:Kingdom_of_Prussia	owl:sameAs	<http://fr.dbpedia.org/resource/Royaume_de_Prusse>
:Albert_Einstein	dbpedia2:workplaces	:University_of_Zurich	owl:sameAs	<http://ja.dbpedia.org/resource/チューリッヒ大学>
:Albert_Einstein	dbpedia:ontology/birthPlace	:German_Empire	<http://purl.org/dc/terms/subject>	:Category:20th_century_in_Germany_by_period
:Albert_Einstein	dbpedia:ontology/knownFor	:Photoelectric_effect	<http://purl.org/dc/terms/subject>	:Category:Electrical_phenomena
:Albert_Einstein	dbpedia:ontology/citizenship	:Switzerland	<http://purl.org/dc/terms/subject>	:Category:Landlocked_countries
:Albert_Einstein	dbpedia:ontology/residence	:Switzerland	<http://purl.org/dc/terms/subject>	:Category:Central_European_countries
:Albert_Einstein	dbpedia:ontology/doctoralAdvisor	:Alfred_Kleiner	<http://purl.org/dc/terms/subject>	:Category:Albert_Einstein
:Albert_Einstein	dbpedia:ontology/doctoralAdvisor	:Alfred_Kleiner	<http://purl.org/dc/terms/subject>	:Category:Swiss_physicists
:Albert_Einstein	dbpedia:ontology/citizenship	:Weimar_Republic	<http://purl.org/dc/terms/subject>	:Category:1910s_in_Germany
:Albert_Einstein	dbpedia2:education	:ETH_Zurich	<http://purl.org/dc/terms/subject>	:Category:1854_establishments_in_Switzerland
:Albert_Einstein	dbpedia:ontology/citizenship	:Kingdom_of_Prussia	<http://purl.org/dc/terms/subject>	:Category:States_of_the_North_German_Confederation
:Albert_Einstein	dbpedia:ontology/birthPlace	:Ulm	<http://purl.org/dc/terms/subject>	:Category:Tübingen_(region)
:Albert_Einstein	dbpedia2:education	:University_of_Zurich	<http://purl.org/dc/terms/subject>	:Category:1833_establishments_in_Switzerland
:Albert_Einstein	dbpedia:ontology/field	:Physics	dbpedia:ontology/wikiPageExternalLink	<http://www.getcited.org/pub/102471397>
:Albert_Einstein	dbpedia:ontology/field	:Physics	dbpedia:ontology/wikiPageExternalLink	<http://www.nature.com/naturephysics>
:Albert_Einstein	dbpedia:ontology/field	:Physics	dbpedia:ontology/wikiPageExternalLink	<https://books.google.com/books?id=JAxLVY96sqsC>
:Albert_Einstein	dbpedia:ontology/knownFor	:Special_relativity	dbpedia:ontology/wikiPageExternalLink	<http://lightspeed.sourceforge.net/>
:Albert_Einstein	dbpedia:ontology/citizenship	:Statelessness	dbpedia:ontology/wikiPageExternalLink	<http://www.unhcr.org/statelessness>
:Albert_Einstein	dbpedia:ontology/deathPlace	:Princeton,_New_Jersey	dbpedia:ontology/wikiPageExternalLink	<http://www.princetonnj.gov>
:Albert_Einstein	dbpedia:ontology/award	:Matteucci_Medal	dbpedia:ontology/wikiPageExternalLink	<http://www.accademiaxl.it/en/awards/120-medaglia-matteucci.html>
:Albert_Einstein	dbpedia:ontology/deathPlace	:Princeton,_New_Jersey	foaf:depiction	<http://commons.wikimedia.org/wiki/Special:FilePath/Lower_Pyne_(Princeton).jpg>

Figura 5.9: Estrutura da informação complementar consultada para a entidade foco

Resumidamente, o objetivo deste módulo consiste na construção e execução de consultas *Sparql* à base de dados semânticas e obter toda a informação necessária (Fig. 5.8) sobre as entidades foco da questão.

5.1.0.6 Geração de resposta

Nesta fase a base de dados local contém toda a informação necessária para responder à questão do utilizador. Assim, é necessário de construir uma consulta *Sparql* final que agrega toda informação resultante dos módulos anteriores, nomeadamente, tipo de resposta, entidades foco e as suas propriedades.

Contudo, antes de efetuar esta consulta final, é necessário de identificar os possíveis predicados existentes na base de dados, ou seja, é necessário de encontrar a correspondência entre os *synsets* da lista de propriedades e os predicados da entidade foco.

Assim, é percorrida a lista de possíveis propriedades e é construída a consulta *Sparql* to tipo *ASK* para cada delas. Este tipo de consulta permite identificar se existe ou não correspondência entre a propriedade e o predicado, e em caso afirmativo, o predicado é acrescentado a lista de possíveis predicados (Fig. 5.10). Com isto, efetua-se uma translação de propriedades para as propriedades semânticas (predicados).

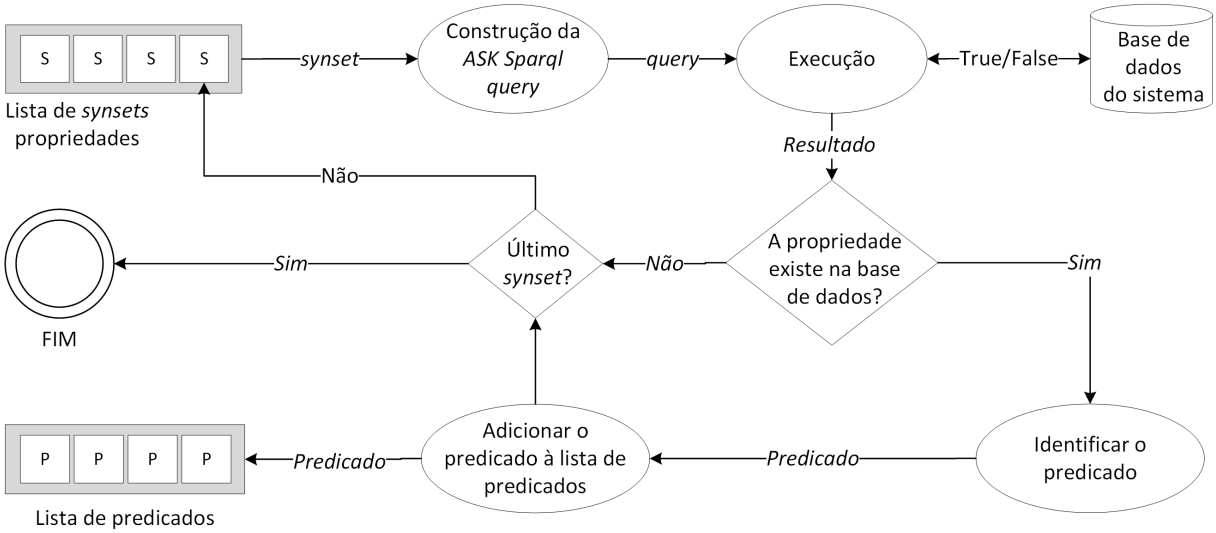


Figura 5.10: Processo de identificação de possíveis predicados

Após a identificação de possíveis predicados, a consulta final pode ser construída e executada à base de dados local, e consiste em seguintes fases:

- Seleção de todos os triplos que têm como sujeito entidade foco da questão
- Restringir os resultados consoante o tipo de respostas esperado
- Restringir os resultados consoante a lista de possíveis predicados

A consulta é efetuada à base de dados local e permite obter um conjunto de triplos, que visam responder à questão do utilizador.

5.1.1 Avaliação

5.1.1.1 Método de Avaliação

Para a avaliação do sistema foi utilizado um conjunto de perguntas factuais, de listagem e de definição, visto que a resposta será pesquisada na base de dados semântica como *DBpedia* que contém informação factual. Tendo em conta que para algumas entidades

existe mais informação do que para outras, as perguntas escolhidas são de diferentes áreas, permitindo assim avaliar a performance do sistema de melhor forma.

1. Qual é a população de Aveiro?
2. Qual é a profundidade do Mar Mediterrâneo?
3. Qual é a profundidade do Oceano Atlântico?
4. Qual é o comprimento da Muralha da China?
5. Quem é o presidente de Portugal?
6. Quem é o presidente de Estados Unidos da América?
7. Quem fundou o Partido Socialista?
8. Quem fundou o Partido Socialista Português?
9. Quem fundou a Porsche?
10. Quando terminou a Ditadura?
11. Quando começou o Europeu de Futebol de 2016?
12. Quando nasceu o Albert Einstein?
13. Onde fica a Ria de Aveiro?
14. Onde fica a Pateira de Fermentelos?
15. O que é um Moliceiro?
16. O que é um Kayak?
17. Quem é o Secretário Geral da ONU?
18. Qual é a velocidade da luz?
19. Qual é a altura do Michael Phelps?
20. Quando nasceu o Albert Einstein?
21. Onde nasceu o Albert Einstein?
22. Quantos golos marcou o Cristiano Ronaldo?

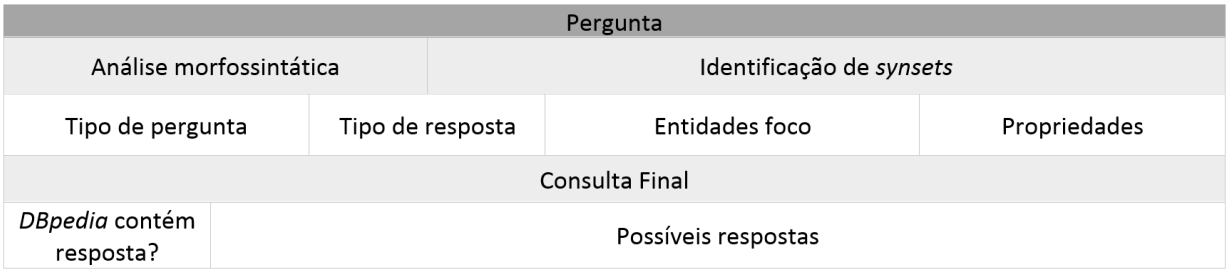


Figura 5.11: Estrutura da tabela de resultados

5.1.1.2 Resultados

Os resultados obtidos para algumas das perguntas referidas anteriormente estão organizados nas tabelas (5.12, 5.13 e 5.14), onde cada coluna (Fig. 5.11) diz respeito a uma análise especifica de cada módulo.

Relativamente ao significado das colunas:

Análise morfossintática - Apresenta o lema e a classe gramatical dos *tokens* que constituem a frase.

Identificação de *synsets* - Apresenta os *synsets* obtidos para todos os *tokens* menos aquele que identifica o tipo de pergunta, visto que serve só para este efeito. Para além dos *synsets* é ainda consultado o seu tipo, podendo ser *Conceito* ou *Entidade Mencionada*.

Tipo de pergunta - Apresenta o tipo de pergunta identificado, já visto anteriormente (5.1).

Tipo de resposta - Apresenta o tipo de dados ou a classe semântica do tipo de resposta, também já consultado anteriormente (Fig. 5.1).

Entidades foco - Representam a lista de *synsets* dos *tokens* que foram identificados como foco da questão.

Propriedades - Apresenta os *synsets* dos *tokens* que foram considerados como propriedades da entidade foco da questão.

Consulta final - Demonstra a consulta SPARQL que agrega toda a informação necessária para a resposta à questão, ou seja, o resultado da sua execução é a resposta à questão.

DBpedia contém resposta? - Resposta pode ser sim ou não, consoante a presença ou ausência de informação necessária para a resposta à questão.

Possíveis respostas - Como o próprio nome indica, apresenta a/as possíveis respostas à questão.

Quando nasceu o Albert Einstein?

LEMA	CLASSE
quando	Adverbio
nascer	Verbo
albert_einstein	Substantivo

LEMA	*SYNSET*	TIPO
nascer	be_born	Conceito
	birth	Conceito
	nativity	Conceito
albert_einstein	Albert_Einstein	Entidade Nomeada

WHEN	xsd:date	Albert_Einstein	• be_born • birth • nativity

```
SELECT * WHERE {
  <http://dbpedia.org/resource/Albert_Einstein> ?p ?o .
  OPTIONAL { ?o rdf:type ?otype. }
  FILTER ( datatype(?o) = xsd:date)
  FILTER (?p = <http://dbpedia.org/ontology/birthDate>  ||
          ?p = <http://dbpedia.org/ontology/birthPlace>) .
}
```

SIM	(?p = <http://dbpedia.org/ontology/birthDate>) (?o = "1879-03-14"^^xsd:date)

Figura 5.12: Exemplo do resultado de uma pergunta factual

O exemplo apresentado na (Fig. 5.12) demonstra a análise pormenorizada de uma pergunta factual *Quando nasceu o Albert Einstein?*.

Após a análise morfossintática obteve-se 3 *tokens*, o seu lema e a classe gramatical, podendo dizer que a frase é constituída por um advérbio, verbo e substantivo, sendo uma estrutura comum de questões factuais.

A seguir é possível observar os *synsets* e o seu tipo, identificados para cada um dos *tokens* anteriores, ignorando o advérbio que só é útil para a identificação do tipo de pergunta. Com esta análise obteve-se 3 *synsets* para o verbo *nascer* e um *synset* para o substantivo *Albert Einstein*, em que o último é do tipo *Entidade Mencionada*, ou seja, é diretamente colocado na lista de entidades foco da frase de acordo com o algoritmo de sistema, e os restantes *synsets* são colocados na lista de propriedades.

Utilizando o advérbio *Quando*, o tipo de pergunta foi classificado como *WHEN* e o tipo de resposta corresponde ao tipo de dados *xml* (*xsd:date*).

Tendo identificada a entidades foco da questão, é construída a consulta à DBpedia para obter toda a informação necessária sobre a entidade foco ($Albert_Einstein$).

Os triplos resultantes da consulta são armazenados na base de dados local e é percorrida a lista de possíveis propriedades para identificar os predicados para a entidade foco. Esta análise identificou dois possíveis predicados *birthDate* e *birthPlace*, visto que ambos predicados contêm a consequência de letras *birth*.

Na fase de construção da consulta *Sparql* final, o sistema já possui informação suficiente para este efeito, nomeadamente, tipo de resposta, entidade foco e os possíveis predicados, resultando na consulta apresentada no respetivo campo.

O último campo apresenta a resposta única e completamente correta. No triplo é possível observar que o *objeto* é do tipo *date* e o predicado é *birthDate*, o que consta na consulta efetuada.

Assim conclui-se que o Albert Einstein nasceu no dia 14 de Março de 1879. A segunda pergunta (Fig. 5.13), *Qual é a altura do Michael Phelps?*, apesar de ser semelhante à per-

Qual é a altura do Michael Phelps?

LEMA	CLASSE
qual	Advérbio
altura	Substantivo
michael_phelps	Substantivo

LEMA	*SYNSET*	TIPO
altura	altura_(castro_marim)	Entidade Nomeada
	altura,_minnesota	Entidade Nomeada
	krull_dimension	Conceito
	altura_(revista)	Entidade Nomeada
	altura_(astronomia)	Conceito
	altura	Entidade Nomeada
	pitch	Conceito
	height	Conceito
michael_phelps	michael_phelps	Entidade Nomeada

WHICH	• xsd:date • dbc:Dog_breeds • dbo:Eukaryote • dbo:Person • owl:Thing • dbo:Place • xsd:time • dbo:Game	• Altura_(Castro_Marim) • Altura,_Minnesota • Altura_(revista) • Altura • Michael_Phelps	• Krull_dimension • Altura_(astronomia) • pitch • height

```
SELECT * WHERE {
  <http://dbpedia.org/resource/Michael_Phelps> ?p ?o .
  OPTIONAL { ?o rdf:type ?otype. }
  FILTER ( datatype(?o) = xsd:date     || ?otype = dbc:Dog_breeds   ||
           ?otype = dbo:Eukaryote      || datatype(?o) = xsd:double ||
           datatype(?o) = xsd:float    || ?otype = dbo:Person       ||
           ?otype = dbo:Place          || datatype(?o) = xsd:time   ||
           datatype(?o) = xsd:integer  || ?otype = dbo:Game         ||
           datatype(?o) = xsd:long)
  FILTER (?p = <http://dbpedia.org/ontology/height>).
}
```

SIM	(?p = <http://dbpedia.org/ontology/height>) (?o = 1.8288)

Figura 5.13: Exemplo do resultado de uma pergunta factual.

---

gunta anterior tem ligeiras diferenças. Em primeiro lugar as perguntas do tipo *WHICH* têm como tipo de resposta um vasto número de tipos, o que não permite identificar concretamente o tipo de resposta.

Neste caso o número de entidades mencionadas nesta pergunta é maior. É possível observar que os termos *altura* e *Michael_Phelps* foram classificados como substantivos e ambos possuem *synsets* do tipo entidade mencionada, o que torna a lista de entidades foco bastante extensa. A análise anterior levou à identificação de 5 entidades nomeadas na frase, ou seja, existem 5 entidades foco e 4 propriedades possíveis. Apesar da existência de múltiplas entidades foco, apenas se conseguiu encontrar a existência do predicado *height* relacionado com o Michael_Phelps, resultando na resposta única e correta. Conclui-se assim que a resposta à questão foi conseguida devido à existência do predicado *height*, visto que o tipo de pergunta não identifica de forma restrita o tipo de dados do objeto. Pode-se assim dizer que de acordo com o DBpedia o Michael Phelps tem a altura de 1.82 metros. *Quantos golos marcou o Cristiano Ronaldo?* (Fig. 5.14) é semelhante à pergunta anterior de acordo com o tipo de resposta, que também consiste num vasto conjunto de tipos. O processo de identificação de *synsets* resultou na identificação de duas entidades foco e uma vasta lista de propriedades e a resposta consiste numa lista onde os predicados não têm um significado explícito apesar de responderem à questão.

Quantos golos marcou o Cristiano Ronaldo?

LEMA	CLASSE
quanto	Advérbio
golo	Substantivo
cristiano_ronaldo	Entidade Nomeada
marcar	Verbo

LEMA	*SYNSET*	TIPO
golo	golo	Conceito
cristiano_ronaldo	cristiano_ronaldo	Entidade nomeada
marcar	clock_in	Conceito
	print	Conceito
	marking	Conceito
	marker	Conceito
	mark_off	Conceito
	crisscross	Conceito
	mark	Conceito

HOWMUCH

- xsd:duration
- xsd:unsignedInt
- xsd:double
- xsd:unsignedShort
- xsd:negativeInteger
- xsd:float
- xsd:integer
- xsd:positiveInteger
- xsd:long
- xsd:unsignedLong

- Cristiano_Ronaldo
- Golo_(river)

- clock_in
- print
- goal
- marking
- marker
- mark_off
- crisscross
- mark

```
SELECT * WHERE {
  <http://dbpedia.org/resource/Cristiano_Ronaldo> ?p ?o .
  OPTIONAL { ?o rdf:type ?otype. }
  FILTER ( datatype(?o) = xsd:duration          || datatype(?o) = xsd:unsignedInt     ||
           datatype(?o) = xsd:double            || datatype(?o) = xsd:unsignedShort   ||
           datatype(?o) = xsd:negativeInteger   || datatype(?o) = xsd:float           ||
           datatype(?o) = xsd:integer           || datatype(?o) = xsd:positiveInteger ||
           datatype(?o) = xsd:long              || datatype(?o) = xsd:unsignedLong)
  FILTER (?p = <http://dbpedia.org/property/goals> || ?p = <http://dbpedia.org/property/nationalgoals>).

}
```

SIM

(?p = <http://dbpedia.org/property/goals>) (?o = 84)
(?p = <http://dbpedia.org/property/goals>) (?o = 3)
(?p = <http://dbpedia.org/property/goals>) (?o = 248)
(?p = <http://dbpedia.org/property/nationalgoals>) (?o = 1)
(?p = <http://dbpedia.org/property/nationalgoals>) (?o = 5)
(?p = <http://dbpedia.org/property/nationalgoals>) (?o = 7)

Figura 5.14: Exemplo do resultado de uma pergunta factual.

5.1.1.3 Resumo dos resultados

A avaliação da primeira variante do sistema demonstrou a capacidade de análise da pergunta do utilizador, identificação de diferentes entidades na frase e entidades foco, sobre as quais foi efetuada a consulta de toda a informação necessária para a resposta.

Os resultados gerais (Tabela: 5.1) de avaliação foram divididos em sete tópicos:

- Número de perguntas em quais as entidades mencionadas ou conceitos foram mal identificados ou não identificados.

- Número de perguntas em quais a consulta de *synsets* não retornou resultados, ou nos casos que os *synsets* obtidos não permitiram chegar à resposta final.

- Número de perguntas para quais não foi possível obter resposta devido a falta de informação no DBpedia.

- Número de perguntas para quais a resposta poderia ser encontrada na descrição da entidade.

- Número de perguntas para as quais não se obteve resposta devido a falta da cobertura do tipo de resposta esperado.
- Número de perguntas não respondidas por falha no módulo de PLN.
- Número de perguntas respondidas corretamente de acordo com DBpedia.

Descrição	Nº de perguntas	%
Entidades Mencionadas/Conceitos não identificados	4	18%
Synset mal identificado	1	5%
Ausência de resposta no DBpedia	7	32%
Existência de resposta na descrição da entidade	2	9%
Tipo de resposta insuficiente para a resposta	2	9%
Falta de identificação da entidade foco	1	5%
Questões respondidas	8	36% (*53%)

Tabela 5.1: Resumo dos resultados da variante 1
O valor com (*) corresponde a percentagem de respostas dadas tendo em conta que para 7 questões não existem respostas no *DBpedia*, resultando em 8 das 15 questões respondidas.

5.1.2 Discussão

Apesar de o sistema demonstrar bom comportamento na resposta a questões (53%), houve um conjunto de falhas relacionadas com diferentes módulos que constituem o sistema.

Na tabela 5.1 é possível ver que a quantidade de questões respondidas é semelhante a quantidade de questões não respondidas por causa da ausência de informação na fonte de dados, sendo a maior limitação para o sistema. Observou-se que a informação disponível varia consoante a entidade mencionada ou conceito, ou seja, existem entidades mais divulgadas/conhecidas do que outras e, desta forma, umas possuem mais propriedades do que outras, como por exemplo nas perguntas *Qual é a profundidade do Oceano Atlântico?* e *Qual é a profundidade do Mar Mediterrâneo?* em que o sistema encontrou a resposta para a primeira, uma vez que a entidade *Oceano_Atlântico* possui uma propriedade *depth*, enquanto que a outra (*Mar_Mediterrâneo*) não. Tendo em conta este problema, a eficácia do sistema depende da entidade mencionada/conceito foco da frase.

A identificação de entidades nomeadas e conceitos também falhou em alguns casos, identificando mal ou mesmo não identificando estas entidades, o que levou a falta de resposta. Sendo que houve casos em que a lista de entidades foco contém mais do que uma entidade, resultando assim em maior processamento e a consulta de informação desnecessária.

Outra falha foi identificada na análise da pergunta *Qual é a velocidade da luz?* em que existiu o problema na identificação da entidade foco, devido à existência de dois conceitos na frase, *Velocidade* e *Luz*. O sistema não conseguiu identificar corretamente qual é a entidade foco e quais são as propriedades.

5.2 Variante 2

A primeira variante do sistema apresentou resultados razoáveis ao responder às questões mais simples, sendo que quando são efetuadas perguntas mais complexas existem algumas limitações que levaram ao desenvolvimento da segunda variante, com objetivo de conseguir resolver estas limitações.

5.2.1 Nova forma de determinação do foco

As limitações consistem na correta identificação da entidade foco da questão, visto que existem questões que possuem só uma entidade (questões simples), e várias entidades (questões complexas). A existência de múltiplas entidades foco dificulta o sistema em encontrar sobre o quê o utilizador efetuou a pergunta, assim como leva a maior processamento da informação ao tentar encontrar a resposta para cada uma das entidades foco identificadas.

Para efetuar a correta deteção da entidade foco da questão é necessário recorrer à análise de dependências, que permite obter relações existentes entre diferentes termos da questão.

A ferramenta *Freeling* utilizada no módulo PLN é capaz de efetuar análise deste tipo, mas após algumas tentativas verificou-se que esta análise não traz resultados pretendidos e não permite identificar a verdadeira entidade foco.

Visto que com a utilização da ferramenta *Freeling* e o seu *parser* de dependências não foi possível identificar a verdadeira e única entidade foco da questão, foi necessário o uso do *Freeling* junto com o *Maltparser*.

Como já referido anteriormente, o *Maltparser* constrói a árvore de dependências tendo um *treebank* de entrada, que neste caso tem de ser construído a partir de uma análise morfossintática do *Freeling*.

Para construção do *treebak* para o treino do *MaltParser*, foi tido em conta o *treebank* original (Fig. 5.17) dos dados do bosque e a análise morfossintática do *Freeling* das frases do bosque.

Na linguística um *treebank* é um corpus de texto analisado que anota a estrutura sintática ou semântica da frase.

Os *treebanks* podem ser formatados de várias formas, sendo que foram abordados dois formatos, *CoNLL-X* e *CoNLL-U*. *CoNLL* é um conferência de alto nível, organizada anualmente por *SIGNLL* (*ACL's Special Interest Group on Natural Language Learning*).

A estrutura de dados em formato CoNLL-U (Fig. 5.15) deriva do formato CoNNL-X (Fig. 5.16) utilizado até ao ano 2007, possuindo diferentes campos do que o formato anterior, nomeadamente *ID, FORM, LEMMA, UPOSTAG, XPOSTAG, FEATS, DEPREL, DEPS e MISC*.

Na atual estrutura (Fig. 5.15), o campo *ID* representa os índices de cada palavra, começando por 1. O campo *FORM* representa a forma como as palavras ocorrem na frase, enquanto que o campo *LEMMA* representa o lema das mesmas, estando ainda associado à sua análise morfossintática.

Os campos *UPOSTAG*, *XPOSTAG* e *FEATS* dizem respeito às anotações morfossintáticas das palavras. De forma mais específica, o campo *UPOSTAG* contém a classe sintática

```
1    They     they    PRON    PRP    Case=Nom|Number=Plur                2    nsubj    2:nsubj|4:nsubj
2    buy      buy     VERB    VBP    Number=Plur|Person=3|Tense=Pres     0    root     0:root
3    and      and     CONJ    CC     _                                   4    cc       4:cc
4    sell     sell    VERB    VBP    Number=Plur|Person=3|Tense=Pres     2    conj     0:root|2:conj
5    books    book    NOUN    NNS    Number=Plur                         2    obj      2:obj|4:obj
6    .        .       PUNCT   .      _                                   2    punct    2:punct
```

Figura 5.15: Exemplo de dados em formato CoNLL-U. (Fonte: http://universaldependencies.org/format.html)

```
<?xml version="1.0" encoding="UTF-8"?>
<dataformat name="conllx">
        <column name="ID" category="INPUT" type="INTEGER"/>
        <column name="FORM" category="INPUT" type="STRING"/>
        <column name="LEMMA" category="INPUT" type="STRING"/>
        <column name="CPOSTAG" category="INPUT" type="STRING"/>
        <column name="POSTAG" category="INPUT" type="STRING"/>
        <column name="FEATS" category="INPUT" type="STRING"/>
        <column name="HEAD" category="HEAD" type="INTEGER"/>
        <column name="DEPREL" category="DEPENDENCY_EDGE_LABEL" type="STRING"/>
        <column name="PHEAD" category="IGNORE" type="INTEGER" default="_"/>
        <column name="PDEPREL" category="IGNORE" type="STRING" default="_"/>

</dataformat>
```

Figura 5.16: Formato CoNNL-X. (Fonte: http://www.maltparser.org)

---

de acordo com a lista *Universal POS tags* (Tabela 2.1), enquanto que o campo *XPOSTAG* opcionalmente contém anotações sobre a classe sintática específica da língua. O campo *FEATS* representa a lista de características morfológicas separada pelo símbolo "|"de acordo com *Universal feature inventory* (Tabela. 5.2), que corresponde a um conjunto de propriedades adicionais lexicais e gramaticais das palavras, que não são cobertas pela classe sintática da palavra.

Por outro lado, os campos *HEAD* e *DEPREL* dizem respeito à anotação sintática e são utilizados para representar a árvore de dependências entre as palavras. Além disso, estes definem as dependências básicas que devem ser estritamente uma árvore.

A representação de dependência avançada, que em geral é um grafo e não uma árvore, é especificada no campo *DEPS*, usando uma lista de pares de relação de *HEAD*. Usam-se dois pontos (:) para separar o *HEAD* e a relação, a barra vertical (|) para separar os itens da lista, e o sublinhado para a lista vazia.

No capitulo 3, foi falado sobre a fácil gestão do *Freeling*, porque permite introduzir a configuração do utilizador. Nesta fase, está vantagem foi aproveitada e algumas alterações foram introduzidas ao módulo de análise morfossintática.

A informação presentes nas colunas 2 (Form), 3 (Lemma), 4 (CPostag) e 6 (Feats) da figura 5.17, pode ser produzida utilizando o *Freeling*, mas com algumas alterações. As colunas 4 e 6 são especificas deste formato de dados, e não são produzidos com análise do

Propriedade	Descrição	Propriedade	Descrição
Abbr	Abreviação	Animacy	Animacidade
Aspect	Aspeto gramatical	Case	Caso
Definite	definiteness or state	Degree	degree of comparison
Evident	evidentiality	Foreign	É uma palavra estrangeira?
Gender	Género	Mood	mood
NumType	Tipo numérico	Number	Número
Number	Número	Person	Pessoa
Polarity	polarity	Polite	politeness
Poss	possessive	PronType	pronominal type
Reflex	reflexive	Tense	tense
VerbForm	Forma da palavra or deverbative	Voice	Voz

Tabela 5.2: Propriedades lexicais adicionais (Fonte: http://universaldependencies.org/u/feat/index.html)

Freeling, enquanto as restantes cumprem a mesma forma de representação e não necessitam de processamento extra. Desta forma, as alterações introduzidas ao *Freeling* consistiram na produção do conteúdo das colunas 4 e 6 a partir da análise original do *Freeling*, que difere na sua forma de representação, mas produz informação suficiente para se poder obter o mesmo formato representaod na figura 5.17.

```
# text = «Orelhas» para os computadores
# source = CETEMPúblico n=3 sec=clt-soc sem=92b
# newdoc id = CP3
# sent_id = CP3-1
# id = 13
1       «               «               PUNCT   PU|@PU                  _                                                       2   punct   _       SpaceAfter=No
2       Orelhas         orelha          NOUN    <np-idf>|N|F|P|@NPHR    Gender=Fem|Number=Plur                                  0   root    _       SpaceAfter=No
3       »               »               PUNCT   PU|@PU                  _                                                       2   punct   _       _
4       para            para            ADP     PRP|@N<                 _                                                       6   case    _       _
5       os              o               DET     <artd>|ART|M|P|@>N      Definite=Def|Gender=Masc|Number=Plur|PronType=Art        6   det     _       _
6       computadores    computador      NOUN    <np-def>|N|M|P|@P<      Gender=Masc|Number=Plur                                  2   nmod    _       _
```

Figura 5.17: Dados do *treebank* original

As restantes colunas permanecem originais, visto que não existe informação suficiente para a sua construção.

5.2.1.1 Treino do *Maltparser*

Para o treino do *Maltparser* foi necessária a contrução do novo *treebank*, ou seja, uma junção do resultado da análise do *Freeling* e dados originais do bosque (Fig. 5.17).

A construção do *treebank* de treino foi divida em duas fases principais, extração e análise das perguntas presentes no *treebank* original e junção com dados originais.

Os dados originais são fornecidos num ficheiro de texto que foi analisado e mais de 3000 perguntas foram extraídas e analisadas, preenchendo assim as colunas: 2,3,4 e 6.

As restantes colunas permaneceram e uma junção de dados foi efetuada, resultando assim num *treebank* de treino (Fig. 5.18) para o *MaltParser*. É possível notar a ausência

de qualquer informação na coluna 5, isto porque, a mesma não pode ser produzida com o *Freeling* e também não afeta o resultado final.

```
# text = «Orelhas» para os computadores
# source = CETEMPúblico n=3 sec=clt-soc sem=92b
# newdoc id = CP3
# sent_id = CP3-1
# id = 13
1       «               «               PUNCT   _       _                                                       2       punct   _       SpaceAfter=No
2       Orelhas         orelha          NOUN    _       Gender=Fem|Number=Plur                                  0       root    _       SpaceAfter=No
3       »               »               PUNCT   _       _                                                       2       punct   _       _
4       para            para            ADP     _       _                                                       6       case    _       _
5       os              o               DET     _       Definite=Def|Gender=Masc|Number=Plur|PronType=Art        6       det     _       _
6       computadores    computador      NOUN    _       Gender=Masc|Number=Plur                                  2       nmod    _       _
```

Freeling

Figura 5.18: *Treebank* para o treino do *Maltparser*

5.2.2 Regras de identificação da entidade foco

Na identificação da entidade foco é necessário analisar as relações existentes entre os termos da questão. As colunas 7 e 8 fornecem informação suficiente para efetuar esta análise e identificar a entidade foco da frase, em que na coluna 7 são apresentados os identificadores dos termos com os quais têm a relação e na coluna 8 é apresentado o tipo de relação universal de dependência.

Na figura 5.19 é apresentado o resultado da análise do *Maltparser* previamente treinado, de uma questão (*Qual é a velocidade de luz?*).

```
1,      Qual,           qual,           PRON,   _,      Gender=Com|Number=Sing,                          4,      nsubj
2,      é,              ser,            VERB,   _,      Mood=Ind|Tense=Pres|Person=3|Number=Sing,        4,      cop
3,      a,              o,              DET,    _,      Gender=Fem|Number=Sing,                          4,      det
4,      velocidade,     velocidade,     NOUN,   _,      Gender=Fem|Number=Sing,                          0,      root
5,      de,             de,             ADP,    _,      _,                                              7,      case
6,      a,              o,              DET,    _,      Gender=Fem|Number=Sing,                          7,      det
7,      luz,            luz,            NOUN,   _,      Gender=Fem|Number=Sing,                          4,      obl
8,      ?,              ?,              Fp,     _,      _,                                              4,      punct
```

Figura 5.19: Exemplo da análise de dependências

Com os resultados obtidos após a análise de um conjunto de questões (Fig. 5.20), verificou-se a existência de uma relação entre diferentes resultados e posteriormente definiu-se um procedimento de identificação da entidade foco da questão.

Na figura 5.20 é possível observar linhas de duas cores, vermelho que identifica o *root* da questão e, verde identifica a entidade foco.

Entre as 5 perguntas presentes na figuras, pode se estabelecer uma relação, ou seja, 4 das 5 perguntas contêm um tipo de relação *root* e *obj* ou *obl*.

O tipo de relação *obj* (*Object*) em muitos casos é a frase nominal que denota a entidade atuada ou que sofre uma mudança de estado ou movimento .

O tipo de relação *obl* (*Oblique nominal*) é usada para um nome nominal (pronome, substantivo) funcionando como um argumento não-núcleo (oblíquo) e o tipo de relação *root* é a relação gramatical raiz que aponta para a raiz da questão.

Ainda é possível observar que, todos os *tokens* marcados com *obj* e *obl*, dependem do *root* da questão. Na terceira pergunta é possível ver que não existe *tokens* com marca *obj* ou *obl*, visto que existe uma única entidade na frase. Contudo, desenvolveu-se uma

sequencia de tarefas que permita identificar a entidade foco da questão de uma forma automática (Fig. 5.21).

5.2.3 Avaliação

O sistema foi avaliado usando o mesmo conjunto de questões utilizado para a avaliação da primeira variante, com o objetivo de verificar as melhorias obtidas.

5.2.3.1 Resultados

Os resultados obtidos demonstraram uma melhoria no tratamento de questões e uma correta identificação da entidade foco.

Na tabela 5.3 é possível observar que o número de questões respondidas aumentou, devido as melhorias efetuadas tanto na parte de processamento de linguagem natural, como na parte de filtragem por tipo de resposta esperado.

Descrição	Nº de perguntas	%
Entidades Mencionadas/Conceitos não identificados	3	14%
Synset mal identificado	1	5%
Ausência de resposta no DBpedia	7	32%
Existência de resposta na descrição da entidade	2	9%
Tipo de resposta insuficiente para a resposta	0	0%
Falta de identificação da entidade foco	0	0%
Questões respondidas	10	45% (*67%)

Tabela 5.3: Resumo dos resultados da variante 2
O valor com (*) corresponde a percentagem de respostas dadas tendo em conta que para 7 questões não existem respostas no *DBpedia*, resultando em 10 das 15 questões respondidas.

5.2.4 Discussão

O desenvolvimento da segunda variante do sistema demonstrou bons resultados, visto que se conseguiu obter a única e correta entidade foco da frase, o que não era possível na primeira variante do sistema. Apesar da identificação da verdadeira entidade foco não ter aumentado consideravelmente o número de perguntas respondidas, reduziu significativamente o tempo de resposta e a quantidade de processamento extra desnecessário. Além disso, esta é uma tarefa essencial no processamento de questões que possuam mais do que uma entidade mencionada ou conceito.

O módulo de processamento de linguagem natural sofreu algumas melhorias comparativamente à primeira variante do sistema, conseguindo identificar mais algumas entidades mencionadas.

Apesar das melhorias efetuadas na segunda variante, o problema de ausência de informação no *DBpedia* permaneceu e não pôde ser resolvido, mas tem uma possível resolução.

Comparativamente à primeira variante do sistema, a quantidade de informação consultada e armazenada reduziu significativamente, devido à existência de uma única entidade foco por questão, enquanto que na primeira variante este número atingia 10 unidades, o que fazia consultar toda a informação necessária para resposta sobre as entidades que não eram o correto foco da questão.

Na tabela 5.4 é possível observar o número total de entidades foco identificadas na primeira e segunda variante do sistema, e observar a significativa diferença, demonstrando assim que a identificação da correta entidade foco da frase é uma análise necessária.

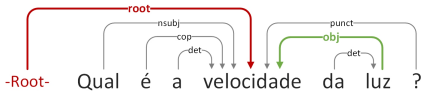
Variante	Nº total de entidades foco resultantes
1	45
2	34

Tabela 5.4: Resultados obtidos com a identificação correta da entidade foco

É possível concluir que a correta identificação da entidade foco da frase permitiu reduzir o número de possíveis entidades foco em 11 unidades (24%). Desta forma, o sistema efetua menos 11 consultas à fonte de dados externa, reduzindo significativamente o tamanho da base de dados local e o tempo de resposta.

```
Qual é a velocidade da luz?

1,      Qual,           qual,           PRON,   _,      Gender=Com|Number=Sing,                          4,      nsubj
2,      é,              ser,            VERB,   _,      Mood=Ind|Tense=Pres|Person=3|Number=Sing,        4,      cop
3,      a,              o,              DET,    _,      Gender=Fem|Number=Sing,                          4,      det
4,      velocidade,     velocidade,     NOUN,   _,      Gender=Fem|Number=Sing,                          0,      root
5,      de,             de,             ADP,    _,      _,                                              7,      case
6,      a,              o,              DET,    _,      Gender=Fem|Number=Sing,                          7,      det
7,      luz,            luz,            NOUN,   _,      Gender=Fem|Number=Sing,                          4,      obl
8,      ?,              ?,              Fp,     _,      _,                                              4,      punct
```

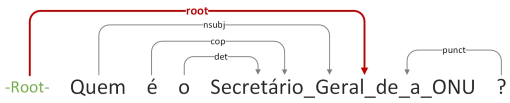



```
Onde fica a Ria de Aveiro?

1,      Onde,           onde,           ADV,    _,      _,                                              2,      nsubj
2,      fica,           ficar,          VERB,   _,      Mood=Ind|Tense=Pres|Person=3|Number=Sing,        0,      root
3,      a,              o,              DET,    _,      Gender=Fem|Number=Sing,                          4,      det
4,      Ria_de_Aveiro,  ria_de_aveiro,  NOUN,   _,      neclass=organization,                           2,      obj
5,      ?,              ?,              Fp,     _,      _,                                              2,      punct
```


```
Quem é o Secretário Geral da ONU?

1,      Quem,           quem,           PRON,   _,      Gender=Com|Number=Inv,                           4,      nsubj
2,      é,              ser,            VERB,   _,      Mood=Ind|Tense=Pres|Person=3|Number=Sing,        4,      cop
3,      o,              o,              DET,    _,      Gender=Masc|Number=Sing,                         4,      det
4,      Secretário_Geral_de_a_ONU,      secretário_geral_de_a_onu,      NOUN,   _,      neclass=Person, 0,      root
5,  ?,                  ?,              Fp,     _,      _,                                              4,      punct
```




```
Qual é a altura do Michael Phelps?

1,      Qual,           qual,           PRON,   _,      Gender=Com|Number=Sing,                          4,      nsubj
2,      é,              ser,            VERB,   _,      Mood=Ind|Tense=Pres|Person=3|Number=Sing,        4,      cop
3,      a,              o,              DET,    _,      Gender=Fem|Number=Sing,                          4,      det
4,      altura,         altura,         NOUN,   _,      Gender=Fem|Number=Sing,                          0,      root
5,      de,             de,             ADP,    _,      _,                                              7,      case
6,      o,              o,              DET,    _,      Gender=Masc|Number=Sing,                         7,      det
7,      Michael_Phelps, michael_phelps, NOUN,   _,      neclass=organization,                           4,      obl
8,      ?,              ?,              Fp,     _,      _,                                              4,      punct
```


```
Qual é a profundidade do Mar Mediterrâneo?

1,      Qual,           qual,           PRON,   _,      Gender=Com|Number=Sing,                          4, nsubj
2,      é,              ser,            VERB,   _,      Mood=Ind|Tense=Pres|Person=3|Number=Sing,        4, cop
3,      a,              o,              DET,    _,      Gender=Fem|Number=Sing,                          4, det
4,      profundidade,   profundidade,   NOUN,   _,      Gender=Fem|Number=Sing,                          0, root
5,      de,             de,             ADP,    _,      _,                                              7, case
6,      o,              o,              DET,    _,      Gender=Masc|Number=Sing,                         7, det
7,      Mar_Mediterrâneo,       mar_mediterrâneo,       NOUN,   _,      neclass=location,               4, obl
8,      ?,              ?,              Fp,     _,      _,                                              4, punct
```

Figura 5.20: Exemplo do resultado da análise de dependências

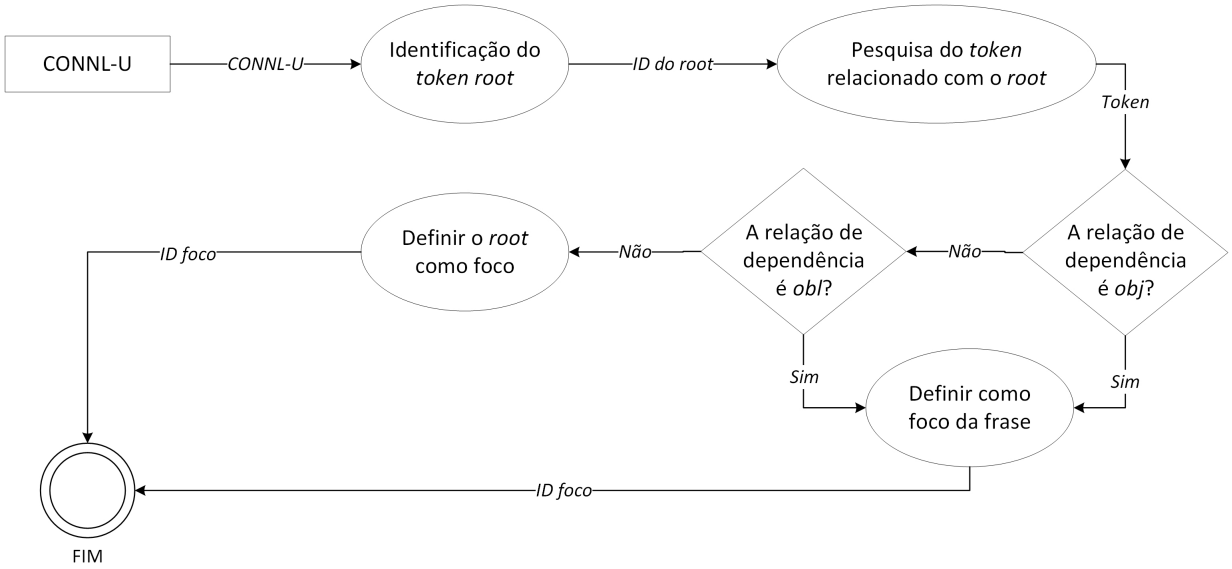


Figura 5.21: Identificação da entidade foco tendo em conta a árvore de dependências.

Capítulo 6

Conclusões

6.1 Resumo do trabalho realizado

Em primeiro lugar foi efetuada uma pesquisa sobre a área de processamento de linguagem natural, visto ser uma área nova.

Uma das principais características sobre a fase de processamento de linguagem natural foi a identificação da intenção do utilizador, nomeadamente, identificação do tipo de pergunta, o tipo de resposta e entidade foco da questão.

Foram identificados diferentes tipos tanto de perguntas como de respostas, dos quais foram selecionados os tipos de perguntas factuais, de listagem e de definição. Os tipos de resposta estão diretamente relacionados com o tipo de pergunta, sendo que representam informação semântica, ou seja, o tipo de resposta apresenta uma classe semântica ou um tipo de dados.

A identificação da intenção do utilizador foi uma das principais tarefas abordadas neste trabalho, o que exigiu uma forte análise da questão do utilizador, identificando diferentes tipos de entidades mencionadas e conceitos na frase, junto com a análise de dependências entre os termos, descobrindo deste modo o foco da questão.

A identificação do foco da questão foi implementada desde a primeira variante do sistema mas, devido ao fraco algoritmo, não apresentou resultados desejáveis, o que levou a uma forte reestruturação na segunda variante do sistema, implementando a análise de dependências, o que permitiu a correta identificação da entidade foco e, não menos importante, reduziu significativamente o custo de processamento e o tamanho de informação consultada para a resposta à questão.

Após a fase de processamento de linguagem natural, foram desenvolvidas técnicas de tradução da consulta do utilizador para a linguagem semântica, visto que a informação para a resposta é consultada numa base de dados semântica. Desta forma, esta fase consistiu em identificar a representação das palavras importantes da frase (verbos, entidades mencionadas ou conceitos) no mundo semântico, nomeadamente, identificar os *synsets* dos mesmos.

Esta fase permitiu então identificar a representação semântica da entidade foco e as

suas características, efetuando a consulta a *DBpedia* com objetivo de agregar a informação suficiente para a resposta a pergunta efetuada.

A fase final consistiu na elaboração de técnicas automáticas de construção de consultas SPARQL, que tinham como resultado a resposta à questão. Estas técnicas incluem a construção de diferentes consultas *Sparql*, identificação de propriedades existentes na base de dados semântica, criação de múltiplos filtros consoante o tipo de resposta, ou seja, a fase final tinha como objetivo agregar toda a informação derivada conseguindo a resposta final.

6.2 Principais resultados

Os resultados demonstraram que o sistema é capaz de responder à maioria (67%) das questões factuais, de listagem ou de definição, tendo em conta que a resposta exista na fonte de dados.

Foram efetuadas questões sobre diferentes entidades mencionadas e conceitos, com objetivo de identificar quais limitações o sistema apresenta.

Diferentes áreas foram abrangidas e identificou-se que a informação existente na fonte de dados escolhida (DBpedia) varia consoante a entidade ou conceito, o que levou à falta de resposta em alguns casos, sendo que estes não se consideram como uma limitação do sistema, mas sim da fonte de dados.

Para além desta limitação principal, existiram outras como a identificação correta de entidades mencionadas, *synsets* para as mesmas e falhas no tipo de resposta esperado presentes na primeira variante do sistema, que foram aperfeiçoadas na segunda variante obtendo assim melhores resultados, ou seja, 67% de questões respondidas.

E por fim, reduziu-se significativamente o processamento e armazenamento extra existente na primeira variante do sistema, devido a falta de identificação de correta entidade foco da questão.

6.3 Trabalho Futuro

Neste momento o sistema tem vários aspetos que podem ser melhorados, tanto na parte de processamento de linguagem natural como na parte de consulta de informação.

O sistema poderá implementar o *feedback* do utilizador em cada fase de processamento, ou seja, existem casos em que a entidade mencionada é erradamente identificada ou mesmo não identificada, e neste caso poderia ser implementado um controlo extra e melhorar o processamento futuro.

Para além das respostas sucintas a questões, o sistema poderia fornecer alguma informação adicional ou a resposta mais extensa, sendo que isto implicava a utilização de técnicas de extração de informação.

As questões já respondidas podiam ser armazenadas e, desta forma, feito uma questão, a resposta para a mesma seria pesquisada em primeiro lugar na base de dados de respostas

já dadas. Esta abordagem pode tanto trazer vantagens, como a diminuição do tempo de resposta, como também desvantagens, processamento extra na pesquisa da reposta, após a geração da mesma caso não exista e no armazenamento da mesma após a resposta ao utilizador. Para além disso, era necessário obter algum *feedback* do utilizador para se certificar se a resposta dada é correta ou não.

Tendo em conta que o sistema é baseado na ontologia, a utilização de novas fontes de dados é mais simples e poderá ser implementada com objetivo de obter a informação ausente no *DBpedia*.

Para melhor experiência do utilizador o sistema poderia fornecer uma interface web, apresentando respostas curtas junto com a fonte de informação onde a mesma foi consultada.

Bibliografia

Balakrishna, M., Werner, S., Tatu, M., Erekhinskaya, T., and Moldovan, D. (2016). K-extractor: Automatic knowledge extraction for hybrid question answering. In *Semantic Computing (ICSC), 2016 IEEE Tenth International Conference on*, pages 390–391. IEEE.

Berners-Lee, T., Hendler, J., Lassila, O., et al. (2001). The semantic web. *Scientific american*, 284(5):28–37.

Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., and Hellmann, S. (2009). Dbpedia-a crystallization point for the web of data. *Web Semantics: science, services and agents on the world wide web*, 7(3):154–165.

Branco, A., Rodrigues, L., Silva, J., and Silveira, S. (2008). Xisquê: An online qa service for portuguese. In *PROPOR*, volume 8, pages 232–235. Springer.

Breck, E., Burger, J. D., Ferro, L., Hirschman, L., House, D., Light, M., and Mani, I. (2000). How to evaluate your question answering system every day and still get real work done. *arXiv preprint cs/0004008*.

Carvalho, G., De Matos, D. M., and Rocio, V. (2008). Idsay: Question answering for portuguese. In *Workshop of the Cross-Language Evaluation Forum for European Languages*, pages 345–352. Springer.

Damljanovic, D., Agatonovic, M., and Cunningham, H. (2011). Freya: An interactive way of querying linked data using natural language. In *Extended Semantic Web Conference*, pages 125–138. Springer.

Elbedweihy, K., Wrigley, S. N., Ciravegna, F., and Zhang, Z. (2013). Using babelnet in bridging the gap between natural language queries and linked data concepts. In *Proceedings of the 2013th International Conference on NLP & DBpedia-Volume 1064*, pages 62–73. CEUR-WS. org.

Fellbaum, C. (1998). Wordnet: an eletronic lexical database. cambridge, massachusetts, eua.

Gimpel, K., Schneider, N., O'Connor, B., Das, D., Mills, D., Eisenstein, J., Heilman, M., Yogatama, D., Flanigan, J., and Smith, N. A. (2011). Part-of-speech tagging for twitter: Annotation, features, and experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 42–47. Association for Computational Linguistics.

Green Jr, B. F., Wolf, A. K., Chomsky, C., and Laughery, K. (1961). Baseball: an automatic question-answerer. In *Papers presented at the May 9-11, 1961, western joint IRE-AIEE-ACM computer conference*, pages 219–224. ACM.

Guha, R., McCool, R., and Miller, E. (2003). Semantic search. In *Proceedings of the 12th international conference on World Wide Web*, pages 700–709. ACM.

Hirschman, L. and Gaizauskas, R. (2001). Natural language question answering: the view from here. *natural language engineering*, 7(04):275–300.

Hoffart, J., Yosef, M. A., Bordino, I., Fürstenau, H., Pinkal, M., Spaniol, M., Taneva, B., Thater, S., and Weikum, G. (2011). Robust disambiguation of named entities in text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 782–792. Association for Computational Linguistics.

Kumar, A., Irsoy, O., Ondruska, P., Iyyer, M., Bradbury, J., Gulrajani, I., Zhong, V., Paulus, R., and Socher, R. (2016). Ask me anything: Dynamic memory networks for natural language processing. In *International Conference on Machine Learning*, pages 1378–1387.

Lavelli, A., Hall, J., Nilsson, J., and Nivre, J. (2009). Maltparser at the evalita 2009 dependency parsing task. *Proceedings of EVALITA Evaluation Campaign*, 31.

Liddy, E. (2001). Natural language processing. encyclopedia of library and information science. ny. marcel decker.

Mishra, A. and Jain, S. K. (2016). A survey on question answering systems with classification. *Journal of King Saud University-Computer and Information Sciences*, 28(3):345–361.

Murphy, M. (2006). Hyponymy and hyperonymy. pages 446–448.

Navigli, R. and Ponzetto, S. P. (2012). Multilingual wsd with just a few lines of code: the babelnet api. In *Proceedings of the ACL 2012 System Demonstrations*, pages 67–72. Association for Computational Linguistics.

Nivre, J., Hall, J., and Nilsson, J. (2006). Maltparser: A data-driven parser-generator for dependency parsing. In *Proceedings of LREC*, volume 6, pages 2216–2219.

Padró, L. and Stanilovsky, E. (2012). Freeling 3.0: Towards wider multilinguality. In *LREC2012*.

Pasca, M. A. and Harabagiu, S. M. (2001). High performance question/answering. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 366–374. ACM.

Quaresma, P., Quintano, L., Rodrigues, I., Saias, J., and Salgueiro, P. (2004). University of évora in qa@ clef-2004. In *Workshop of the Cross-Language Evaluation Forum for European Languages*, pages 534–543. Springer.

Segaran, T., Evans, C., Taylor, J., Toby, S., Colin, E., and Jamie, T. (2009). *Programming the semantic web*. O'Reilly Media, Inc.

Song, D., Schilder, F., Smiley, C., Brew, C., Zielund, T., Bretz, H., Martin, R., Dale, C., Duprey, J., Miller, T., et al. (2015). TR Discover: A natural language interface for querying and analyzing interlinked datasets. In *International Semantic Web Conference*, pages 21–37. Springer.

Tahri, A. and Tibermacine, O. (2013). Dbpedia based factoid question answering system. *International Journal of Web & Semantic Technology*, 4(3):23.

Tjong Kim Sang, E. F. and De Meulder, F. (2003). Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 142–147. Association for Computational Linguistics.

Unger, C., Bühmann, L., Lehmann, J., Ngonga Ngomo, A.-C., Gerber, D., and Cimiano, P. (2012). Template-based question answering over rdf data. In *Proceedings of the 21st international conference on World Wide Web*, pages 639–648. ACM.

Voutilainen, A. (2003). Part-of-speech tagging. *The Oxford handbook of computational linguistics*, pages 219–232.

Winkler, W. E. (1990). String comparator metrics and enhanced decision rules in the fellegi-sunter model of record linkage.

Yao, X. and Van Durme, B. (2014). Information extraction over structured data: Question answering with freebase. In *ACL (1)*, pages 956–966.

Apêndice A

Análise de todas as questões

Onde nasceu o Albert Einstein?

LEMA	CLASSE
onde	Advérbio
nascer	Verbo
albert_einstein	Substantivo

LEMA	*SYNSET*	TIPO
nascer	be_born	Conceito
	birth	
	nativity	
albert_einstein	Albert_Einstein	Entidade Mencionada

WHERE	• dbo:Place • xsd:float • geo:geometry • geo:lat • geo:long	• Albert_Einstein	• be_born • birth • nativity

```
SELECT * WHERE {
  <http://dbpedia.org/resource/Albert_Einstein> ?p ?o .
  OPTIONAL { ?o rdf:type ?otype. }
  FILTER ( datatype(?o) = xsd:float || ?otype = dbo:Place ||
           ?otype = geo:geometry || ?otype = geo:lat || ?otype = geo:long) .
  FILTER (?p = dbo:birthDate || ?p = dbo:birthPlace).
}
```

SIM	(?p = <http://dbpedia.org/ontology/birthPlace >) (?o = <http://dbpedia.org/resource/Ulm>) (?p = <http://dbpedia.org/ontology/birthPlace >) (?o = <http://dbpedia.org/resource/German_Empire>) (?p = <http://dbpedia.org/ontology/birthPlace >) (?o = <http://dbpedia.org/resource/Kingdom_of_Württemberg>)

Figura A.1: Análise da questão "Onde nasceu o Albert Einstein?"

Onde fica a Pateira de Fermentelos?

LEMA	CLASSE
onde	Advérbio
ficar	Verbo
pateira_de_fermentelos	Substantivo

LEMA	*SYNSET*	TIPO
ficar	dallier	Conceito
	oddment	
	leftover	
	be	
	linger	
	remain	
	lingerer	
pateira_de_fermentelos	Pateira_de_Fermentelos	Entidade Mencionada

WHERE	• dbo:Place • xsd:float • geo:geometry • geo:lat • geo:long	• Pateira_de_Fermentelos	• dallier • oddment • leftover • be • linger • remain • lingerer

NÃO

Figura A.2: Análise da questão "Onde fica a Pateira de Fermentelos?"

Onde fica a Ria de Aveiro?

LEMA	CLASSE
onde	Advérbio
ficar	Verbo
ria_de_aveiro	Substantivo

LEMA	*SYNSET*	TIPO
ficar	dallier	Conceito
	oddment	
	leftover	
	be	
	linger	
	remain	
	lingerer	
ria_de_aveiro	Aveiro_Lagoon	Entidade Mencionada

WHERE	• dbo:Place • xsd:float • geo:geometry • geo:lat • geo:long	• Aveiro_Lagoon	• dallier • oddment • leftover • be • linger • remain • lingerer

```
SELECT * WHERE {
  <http://dbpedia.org/resource/Aveiro_Lagoon> ?p ?o .
  OPTIONAL { ?o rdf:type ?otype. }
  FILTER ( datatype(?o) = xsd:float || ?otype = dbo:Place ||
           ?otype = geo:geometry || ?otype = geo:lat || ?otype = geo:long).
}
```

SIM	(?p = <http://www.w3.org/2003/01/geo/wgs84_pos#lat>) (?o = 40.5989) (?p = <http://www.w3.org/2003/01/geo/wgs84_pos#long>) (?o = -8.74625)

Figura A.3: Análise da questão "Onde fica a Ria de Aveiro?"

O que é um Kayak?

LEMA	CLASSE
kayak	Substantivo

LEMA	*SYNSET*	TIPO
kayak	Kayak	Conceito

DEFINITION

- dbo:abstract
- Kayak

```
SELECT * WHERE {
  <http://dbpedia.org/resource/Kayak> ?p ?o .
  FILTER (?p = dbo:abstract)
  FILTER (lang(?o) = 'en' || lang(?o) = 'pt')
}
```

SIM

(?p = <http://dbpedia.org/ontology/abstract>)

(?o = "A kayak is a small, narrow boat which is propelled by means of a double-bladed paddle. The word kayak originates from the Greenlandic language, where it is the word qajaq (pronounced [qajaq]). In the UK the term canoe is often used when referring to a kayak.The traditional kayak has a covered deck and one or more cockpits, each seating one paddler. The cockpit is sometimes covered by a spray deck that prevents the entry of water from waves or spray and makes it possible for suitably skilled kayakers to roll the kayak: that is, to capsize and right it without it filling with water or ejecting the paddler. Some modern boats vary considerably from a traditional design but still claim the title "kayak", for instance in eliminating the cockpit by seating the paddler on top of the boat ("sit-on-top" kayaks); having inflated air chambers surrounding the boat; replacing the single hull by twin hulls, and replacing paddles with other human-powered propulsion methods, such as foot-powered rotational propellers and "flippers". Kayaks are also being sailed, as well as propelled by means of small electric motors, and even by outboard gas engines. The kayak was first used by the indigenous Aleut, Inuit, Yupik and possibly Ainu hunters in subarctic regions of the world."@en)

(?p = <http://dbpedia.org/ontology/abstract>)

(?o = "O caiaque, ou caíque, é uma pequena embarcação a remos utilizada para lazer, transporte e competições. Na vertente desportiva compreende várias modalidades como velocidade, slalom, adaptada, descida, maratona, oceânica, onda, pólo, rafting e rodeio Esta embarcação começou a ficar famosa a partir da década de 1970, quando os programas de televisão começaram a divulgar os desportos radicais. O caiaque nasceu na Groelândia e existe desde tempos imemoriais, servindo de meio de pesca e trabalho aos esquimós. Caiaque significa na língua local "Barco de Caçador", e seu uso era permitido exclusivamente aos homens, que empregavam ossos de baleia, peles e tripas de focas para a construção dessa curiosa embarcação. Os ossos flexíveis de baleia formavam a estrutura do engenho e nas costuras de pele do revestimento usavam-se as tripas de foca. A impermeabilização era obtida pela imersão do caiaque nas águas do mar, ocasionando o seu encharcamento. Para mantê-lo imune à penetração da água, o recurso era muito simples: bastava imergi-lo sempre que não era usado, daí a se constatar que o fundo do mar foi a primeira "garagem" conhecida dos caiaques. O material usado no início pelos esquimós eram as peles de animais; hoje em dia são utilizados fibra de vidro e plástico. Os caiaques de plástico são ideais para regatas ou "White water", pois têm grande resistência a impactos em pedras. O plástico utilizado normalmente é o polietilieno de média densidade e são fabricados pelo processo da rotomoldagem."@pt)

Figura A.4: Análise da questão "O que é um Kayak?"

O que é um Moliceiro?

LEMA	CLASSE
moliceiro	Substantivo

LEMA	*SYNSET*	TIPO
moliceiro	Moliceiro	Conceito

DEFINITION

- dbo:abstract
- Moliceiro

```
SELECT * WHERE {
  <http://dbpedia.org/resource/Moliceiro> ?p ?o .
  FILTER (?p = dbo:abstract)
  FILTER (lang(?o) = 'en' || lang(?o) = 'pt')
}
```

NÃO

Figura A.5: Análise da questão "O que é um Moliceiro?"

Qual é o comprimento da Muralha da China?

LEMA	CLASSE
qual	Advérbio
comprimento	Substantivo
muralha_de_a_china	Substantivo

LEMA	*SYNSET*	TIPO
comprimento	legth	Conceito
muralha_de_a_china		

WHICH	• xsd:date • dbc:Dog_breeds • dbo:Eukaryote • dbo:Person • owl:Thing • dbo:Place • xsd:time • dbo:Game		• length
NÃO			

Figura A.6: Análise da questão "Qual é o comprimento da Muralha da China?"

Qual é a população de Aveiro?

LEMA	CLASSE
qual	Advérbio
população	Substantivo
aveiro	Substantivo

LEMA	*SYNSET*	TIPO
população	population	Conceito
	População_(biologia)	Conceito
aveiro	Aveiro, Pará	Entidade Mencionada
	Aveiro	Entidade Mencionada

WHICH	• xsd:date • dbc:Dog_breeds • dbo:Eukaryote • dbo:Person • owl:Thing • dbo:Place • xsd:time • dbo:Game	• Aveiro, Pará • Aveiro	• Population • População_(biologia)

```
SELECT * WHERE {
<http://dbpedia.org/resource/Aveiro,_Pará> ?p ?o .
 OPTIONAL { ?o rdf:type ?otype. }
 FILTER ( datatype(?o) = xsd:date || ?otype = dbc:Dog_breeds ||
          ?otype = dbo:Eukaryote || datatype(?o) = xsd:double ||
          datatype(?o) = xsd:float || ?otype = dbo:Person ||
          ?otype = dbo:Place || datatype(?o) = xsd:time ||
          datatype(?o) = xsd:integer || ?otype = dbo:Game ||
          datatype(?o) = xsd:long)
}
```

```
SELECT * WHERE {
<http://dbpedia.org/resource/Aveiro> ?p ?o .
 OPTIONAL { ?o rdf:type ?otype. }
 FILTER ( datatype(?o) = xsd:date || ?otype = dbc:Dog_breeds ||
          ?otype = dbo:Eukaryote || datatype(?o) = xsd:double ||
          datatype(?o) = xsd:float || ?otype = dbo:Person ||
          ?otype = dbo:Place || datatype(?o) = xsd:time ||
          datatype(?o) = xsd:integer || ?otype = dbo:Game ||
          datatype(?o) = xsd:long)
}
```

NÃO

Figura A.7: Análise da questão "Qual é a população de Aveiro?"

Qual é a profundidade do Mar Mediterrâneo?

LEMA	CLASSE
qual	Advérbio
profundidade	Substantivo
mar_mediterrâneo	Substantivo

LEMA	*SYNSET*	TIPO
profundidade	depth	Conceito
mar_mediterrâneo	Mediterranean_Sea	Entidade Mencionada

WHICH	• xsd:date • dbc:Dog_breeds • dbo:Eukaryote • dbo:Person • owl:Thing • dbo:Place • xsd:time • dbo:Game	• Mediterranean_Sea	• depth

```
SELECT * WHERE {
<http://dbpedia.org/resource/Mediterranean_Sea> ?p ?o .
 OPTIONAL { ?o rdf:type ?otype. }
 FILTER ( datatype(?o) = xsd:date || ?otype = dbc:Dog_breeds ||
          ?otype = dbo:Eukaryote || datatype(?o) = xsd:double ||
          datatype(?o) = xsd:float || ?otype = dbo:Person ||
          ?otype = dbo:Place || datatype(?o) = xsd:time ||
          datatype(?o) = xsd:integer || ?otype = dbo:Game ||
          datatype(?o) = xsd:long)
 FILTER (?p = <http://dbpedia.org/ontology/averageDepth> ||
          ?p = <http://dbpedia.org/ontology/maximumDepth>).
}
```

SIM	(?p = <http://dbpedia.org/ontology/averageDepth>) (?o = "1500.0"^^xsd:double) (?p = <http://dbpedia.org/ontology/maximumDepth>) (?o = "5267.0"^^xsd:double)

Figura A.8: Análise da questão "Qual é a profundidade do Mar Mediterrâneo?"

Qual é a profundidade do Oceano Atlântico?

LEMA	CLASSE
qual	Advérbio
profundidade	Substantivo
oceano_atlântico	Substantivo

LEMA	*SYNSET*	TIPO
profundidade	depth	Conceito
oceano_atlântico	Atlantic_Ocean	Entidade Mencionada

WHICH	• xsd:date • dbc:Dog_breeds • dbo:Eukaryote • dbo:Person • owl:Thing • dbo:Place • xsd:time • dbo:Game	• Atlantic_Ocean	• depth

```
SELECT * WHERE {
<http://dbpedia.org/resource/Atlantic_Ocean> ?p ?o .
 OPTIONAL { ?o rdf:type ?otype. }
 FILTER ( datatype(?o) = xsd:date || ?otype = dbc:Dog_breeds ||
          ?otype = dbo:Eukaryote || datatype(?o) = xsd:double ||
          datatype(?o) = xsd:float || ?otype = dbo:Person ||
          ?otype = dbo:Place || datatype(?o) = xsd:time ||
          datatype(?o) = xsd:integer || ?otype = dbo:Game ||
          datatype(?o) = xsd:long)
}
```

NÃO	

Figura A.9: Análise da questão "Qual é a profundidade do Oceano Atlântico?"

Qual é a velocidade da luz?

LEMA	CLASSE
qual	Advérbio
velocidade	Substantivo
luz	Substantivo

LEMA	*SYNSET*	TIPO
velocidade	velocity	Conceito
luz	Luz	Entidade Mencionada
	Luz_(Santa_Cruz_da_Graciosa)	
	Luz,_Minas_Gerais	
	Luz_(álbum_de_Roupa_Nova)	
	Luz_(álbum_de_Novo_Som)	
	Luz_(álbum_de_Pedro_Abrunhosa)	
	Luz_(álbum_de_Djavan)	
	Luz_(canção)	
	Luz_(Mourão)	
	Luz_(Lagos)	
	light	Conceito

WHICH

- xsd:date
- dbc:Dog_breeds
- dbo:Eukaryote
- dbo:Person
- owl:Thing
- dbo:Place
- xsd:time
- dbo:Game

- Atlantic_Ocean

- depth

```
SELECT * WHERE {
<http://dbpedia.org/resource/Atlantic_Ocean> ?p ?o .
 OPTIONAL { ?o rdf:type ?otype. }
 FILTER ( datatype(?o) = xsd:date || ?otype = dbc:Dog_breeds ||
          ?otype = dbo:Eukaryote || datatype(?o) = xsd:double ||
          datatype(?o) = xsd:float || ?otype = dbo:Person ||
          ?otype = dbo:Place || datatype(?o) = xsd:time ||
          datatype(?o) = xsd:integer || ?otype = dbo:Game ||
          datatype(?o) = xsd:long)
}
```

NÃO

Figura A.10: Análise da questão "Qual é a velocidade da luz?"

Qual é a altura do Michael Phelps?

LEMA	CLASSE
qual	Advérbio
altura	Substantivo
michael_phelps	Substantivo

LEMA	*SYNSET*	TIPO
altura	altura_(castro_marim)	Entidade Nomeada
	altura,_minnesota	Entidade Nomeada
	krull_dimension	Conceito
	altura_(revista)	Entidade Nomeada
	altura_(astronomia)	Conceito
	altura	Entidade Nomeada
	pitch	Conceito
	height	Conceito
michael_phelps	michael_phelps	Entidade Nomeada

WHICH	• xsd:date • dbc:Dog_breeds • dbo:Eukaryote • dbo:Person • owl:Thing • dbo:Place • xsd:time • dbo:Game	• Altura_(Castro_Marim) • Altura,_Minnesota • Altura_(revista) • Altura • Michael_Phelps	• Krull_dimension • Altura_(astronomia) • pitch • height

```
SELECT * WHERE {
  <http://dbpedia.org/resource/Michael_Phelps> ?p ?o .
  OPTIONAL { ?o rdf:type ?otype. }
  FILTER ( datatype(?o) = xsd:date     || ?otype = dbc:Dog_breeds   ||
           ?otype = dbo:Eukaryote      || datatype(?o) = xsd:double ||
           datatype(?o) = xsd:float    || ?otype = dbo:Person       ||
           ?otype = dbo:Place          || datatype(?o) = xsd:time   ||
           datatype(?o) = xsd:integer  || ?otype = dbo:Game         ||
           datatype(?o) = xsd:long)
  FILTER (?p = <http://dbpedia.org/ontology/height>).
}
```

SIM	(?p = <http://dbpedia.org/ontology/height>) (?o = 1.8288)

Figura A.11: Análise da questão "Qual é a altura do Michael Phelps?"

Quando terminou a Ditadura?

LEMA	CLASSE
quando	Advérbio
terminar	Verbo
ditadura	Substantivo

LEMA	*SYNSET*	TIPO
terminar	death	Conceito
	end_point	
	break_up	
	conclusion	
	closing	
	finish	
	end	
	finalisation	
	close	
	closedown	
	breach	
	finish_off	
	conclude	
ditadura	authoritarianism	Conceito

WHEN	• dbo:Person • dbo:Agent	• authoritarianism	• death • end_point • break_up • conclusion • closing • finish • end • finalisation • close • closedown • breach • finish_off • conclude
NÃO			

Figura A.12: Análise da questão "Quando terminou a Ditadura?"

Quando começou o Europeu de Futebol de 2016?

LEMA	CLASSE
quando	Advérbio
começar	Verbo
europeu_de_futebol	Substantivo
2016	null

LEMA	*SYNSET*	TIPO
começar	induction	Conceito
	commencement	
	approach	
	originator	
	inception	
	commence	
	beginning	
	father	
europeu_de_futebol		

WHEN	• xsd:date		• commence • beginning • father • inception • originator • approach • commencement • induction

SIM

Figura A.13: Análise da questão "Quando começou o Europeu de Futebol de 2016?"

Quando nasceu o Albert Einstein?

LEMA	CLASSE
quando	Adverbio
nascer	Verbo
albert_einstein	Substantivo

LEMA	*SYNSET*	TIPO
nascer	be_born	Conceito
	birth	Conceito
	nativity	Conceito
albert_einstein	Albert_Einstein	Entidade Nomeada

WHEN	xsd:date	Albert_Einstein	• be_born • birth • nativity

```
SELECT * WHERE {
  <http://dbpedia.org/resource/Albert_Einstein> ?p ?o .
  OPTIONAL { ?o rdf:type ?otype. }
  FILTER ( datatype(?o) = xsd:date)
  FILTER (?p = <http://dbpedia.org/ontology/birthDate>  ||
          ?p = <http://dbpedia.org/ontology/birthPlace>) .
}
```

SIM (?p = <http://dbpedia.org/ontology/birthDate>) (?o = "1879-03-14"^^xsd:date)

Figura A.14: Análise da questão "Quando nasceu o Albert Einstein?"

Quem é o secretário Geral do ONU?

LEMA	CLASSE
quem	Advérbio
secretário	Substantivo
geral_de_o_onu	Substantivo

LEMA	*SYNSET*	TIPO
secretário	genus_Sagittarius	Conceito
	secretary_bird	
	family_Sagittariidae	
geral_de_o_onu		

WHO	• dbo:Person • dbo:Agent		• genus_Sagittarius • secretary_bird • family_Sagittariidae
NÃO			

Figura A.15: Análise da questão "Quem é o secretário Geral do ONU?"

Quem fundou o Partido Socialista?

LEMA	CLASSE
quem	Advérbio
fundar	Verbo
partido_socialista	Substantivo

LEMA	*SYNSET*	TIPO
fundar	found	Conceito
	father	
	organization	
	introduction	
partido_socialista	Partido_Socialista_(Colômbia)	Entidade Mencionada
	Socialist_Party_(Netherlands)	
	Socialist_Party_(France)	
	Socialist_Party_(Portugal)	
	Parti_Socialiste_(Belgium)	
	Partido_Socialista_(Brasil)	

WHO	• dbo:Person • dbo:Agent	• Partido_Socialista_(Colômbia) • Socialist_Party_(Netherlands) • Socialist_Party_(France) • Socialist_Party_(Portugal) • Parti_Socialiste_(Belgium) • Partido_Socialista_(Brasil)	• found • father • organization • introduction

```
SELECT * WHERE {
  <http://dbpedia.org/resource/Socialist_Party_(Portugal)> ?p ?o .
  OPTIONAL { ?o rdf:type ?otype. }
  FILTER ( ?otype = dbo:Agent || ?otype = dbo:Person)
  FILTER ( ?p = <http://dbpedia.org/ontology/found> ||
           ?p = <http://dbpedia.org/ontology/father> ||
           ?p = <http://dbpedia.org/ontology/organization> ||
           ?p = <http://dbpedia.org/ontology/introduction> ||
         ).
}
```

SIM

Figura A.16: Análise da questão "Quem fundou o Partido Socialista?"

Quem fundou o Partido Socialista Português?

LEMA	CLASSE
quem	Advérbio
fundar	Verbo
partido_socialista_português	Substantivo

LEMA	*SYNSET*	TIPO
fundar	found	Conceito
	father	
	organization	
	introduction	
partido_socialista_português	Portuguese_Socialist_Party	Entidade Mencionada

WHO	• dbo:Person • dbo:Agent	• Portuguese_Socialist_Party	• found • father • organization • introduction

```
SELECT * WHERE {
  <http://dbpedia.org/resource/Portuguese_Socialist_Party> ?p ?o .
  OPTIONAL { ?o rdf:type ?otype. }
  FILTER ( ?otype = dbo:Agent || ?otype = dbo:Person)
  FILTER ( ?p = <http://dbpedia.org/ontology/found> ||
           ?p = <http://dbpedia.org/ontology/father> ||
           ?p = <http://dbpedia.org/ontology/organization> ||
           ?p = <http://dbpedia.org/ontology/introduction> ||
         ).
}
```

SIM

Figura A.17: Análise da questão "Quem fundou o Partido Socialista Português?"

Quem fundou a Porsche?

LEMA	CLASSE
quem	Advérbio
fundar	Verbo
porsche	Substantivo

LEMA	*SYNSET*	TIPO
fundar	found	Conceito
	father	
	organization	
	introduction	
porsche	Porsche	Entidade Mencionada

WHO	• dbo:Person • dbo:Agent	• Porsche	• found • father • organization • introduction

```
SELECT * WHERE {
  <http://dbpedia.org/resource/Porsche> ?p ?o .
  OPTIONAL { ?o rdf:type ?otype. }
  FILTER ( ?otype = dbo:Agent || ?otype = dbo:Person)
  FILTER ( ?p = <http://dbpedia.org/property/foundation> ||
           ?p = <http://dbpedia.org/ontology/foundationPlace> ||
           ?p = <http://dbpedia.org/ontology/foundedBy> ||
           ?p = <http://dbpedia.org/ontology/foundingYear> ||
           ?p = <http://dbpedia.org/property/founder> ).
}
```

SIM	(?p = <http://dbpedia.org/ontology/foundedBy>) (?o = <http://dbpedia.org/resource/Ferdinand_Porsche>) (?p = <http://dbpedia.org/property/founder>) (?o = <http://dbpedia.org/resource/Ferdinand_Porsche>)

Figura A.18: Análise da questão "Quem fundou a Porsche?"

Quem é o presidente de Estados Unidos de América?

LEMA	CLASSE
quem	Advérbio
presidente	Substantivo
estados_unidos_de_américa	Substantivo

LEMA	*SYNSET*	TIPO
presidente	Presidente_(província)	Entidade Mencionada
	Presidente_(Imbé)	Conceito
	president	Conceito
estados_unidos_de_américa		

WHO	• dbo:Person • dbo:Agent		• Presidente_(Imbé) • president

SIM

Figura A.19: Análise da questão "Quem é o presidente de Estados Unidos de América?"

Quem é o presidente de Portugal?

LEMA	CLASSE
quem	Advérbio
presidente	Substantivo
portugal	Substantivo

LEMA	*SYNSET*	TIPO
presidente	Presidente_(província)	Entidade Mencionada
	Presidente_(Imbé)	Conceito
	president	Conceito
portugal	Portugal	Entidade Mencionada

WHO	• dbo:Person • dbo:Agent	• Portugal	• Presidente_(Imbé) • president

```
SELECT * WHERE {
  <http://dbpedia.org/resource/Portugal> ?p ?o .
  OPTIONAL { ?o rdf:type ?otype. }
  FILTER ( ?otype = dbo:Agent || ?otype = dbo:Person)
}
```

SIM	(?p = <http://dbpedia.org/ontology/leader>) (?o =< http://dbpedia.org/ontology/resource/Eduardo_Ferro_Rodrigues>) (?p = <http://dbpedia.org/ontology/leader>) (?o =< http://dbpedia.org/ontology/resource/António_Costa>) (?p = <http://dbpedia.org/ontology/leader>) (?o =< http://dbpedia.org/ontology/resource/Marcelo_Rebelo_de_Sousa>)

Figura A.20: Análise da questão "Quem é o presidente de Portugal?"

Quantos golos marcou o Cristiano Ronaldo?

LEMA	CLASSE
quanto	Advérbio
golo	Substantivo
cristiano_ronaldo	Entidade Nomeada
marcar	Verbo

LEMA	*SYNSET*	TIPO
golo	golo	Conceito
cristiano_ronaldo	cristiano_ronaldo	Entidade nomeada
marcar	clock_in	Conceito
	print	Conceito
	marking	Conceito
	marker	Conceito
	mark_off	Conceito
	crisscross	Conceito
	mark	Conceito

HOWMUCH	• xsd:duration • xsd:unsignedInt • xsd:double • xsd:unsignedShort • xsd:negativeInteger • xsd:float • xsd:integer • xsd:positiveInteger • xsd:long • xsd:unsignedLong	• Cristiano_Ronaldo • Golo_(river)	• clock_in • print • goal • marking • marker • mark_off • crisscross • mark

```
SELECT * WHERE {
  <http://dbpedia.org/resource/Cristiano_Ronaldo> ?p ?o .
  OPTIONAL { ?o rdf:type ?otype. }
  FILTER ( datatype(?o) = xsd:duration          || datatype(?o) = xsd:unsignedInt     ||
           datatype(?o) = xsd:double            || datatype(?o) = xsd:unsignedShort   ||
           datatype(?o) = xsd:negativeInteger   || datatype(?o) = xsd:float           ||
           datatype(?o) = xsd:integer           || datatype(?o) = xsd:positiveInteger ||
           datatype(?o) = xsd:long              || datatype(?o) = xsd:unsignedLong)
  FILTER (?p = <http://dbpedia.org/property/goals> || ?p = <http://dbpedia.org/property/nationalgoals>).

}
```

SIM	(?p = <http://dbpedia.org/property/goals>) (?o = 84) (?p = <http://dbpedia.org/property/goals>) (?o = 3) (?p = <http://dbpedia.org/property/goals>) (?o = 248) (?p = <http://dbpedia.org/property/nationalgoals>) (?o = 1) (?p = <http://dbpedia.org/property/nationalgoals>) (?o = 5) (?p = <http://dbpedia.org/property/nationalgoals>) (?o = 7)

Figura A.21: Análise da questão "Quantos golos marcou o Cristiano Ronaldo?"